

APLIKASI PENJUALAN PULSA ELEKTRONIK BERBASIS ANDROID

TUGAS AKHIR

Oleh :

Dwi Teguh Prasetyo 3311401047

Disusun untuk memenuhi syarat kelulusan Program Diploma DIII



**PROGRAM STUDI TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM**

BATAM

2016

HALAMAN PENGESAHAN

APLIKASI PENJUALAN PULSA ELEKTRONIK BERBASIS ANDROID

Disusun oleh :


**Dwi Teguh Prasetyo
3311401047**

Telah diuji dan dipertahankan di depan Tim Penguji
dalam Sidang Tugas Akhir
pada tanggal 10 Januari 2017
dan dinyatakan **LULUS**.

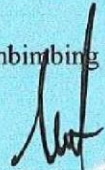
Batam, 12 Januari 2017

Diperiksa dan disetujui oleh;

Pembimbing I,


Nur Zahrafi Janah, S.Kom., M.Sc.
NIK. 112087

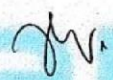
Pembimbing II,


Supardianto, S.ST.
NIK. 113105

Penguji I,


Hilda Widayastuti S.T., M.T.
NIP. 197705122012122001

Penguji II,


Sartikha, S.ST.
NIK. 113115

polibatam

HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311401047

Nama : Dwi Teguh Prasetyo

adalah mahasiswa Teknik Informatika Politeknik Batam yang menyatakan bahwa tugas akhir dengan judul:

Aplikasi Penjualan Pulsa Elektronik Berbasis Android

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 12 Januari 2017

Dwi Teguh Prasetyo
NIM. 3311401047

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul “**Aplikasi Penjualan Pulsa Elektronik Berbasis Android**”. Penulisan laporan ini kami buat untuk memenuhi Tugas Akhir sebagai salah satu syarat mendapatkan gelar Diploma DIII.

Penulis mengucapkan terima kasih kepada semua pihak atas bantuan, nasehat, dukungan, serta bimbingan dalam pembuatan Tugas Akhir ini, sehingga penulis dapat menyelesaikan laporan tugas akhir ini tepat waktu. Dengan kerendahan hati, pada kesempatan ini penulis mengucapkan rasa terima kasih yang sebesar-besarnya kepada :

1. Orang tua yang selalu mendoakan, memberi dukungan dan kasih sayang serta nasehat.
2. Ibu Nur Zahradi Janah, S.Kom., M.Sc dan Bapak Supardianto, S.St selaku dosen pembimbing Tugas Akhir dan Ibu Sartikha, S.St. selaku dosen pembimbing tugas proyek akhir II yang telah memberikan bantuan, nasehat, dan bimbingan yang sangat berguna dalam penyusunan proyek akhir ini.
3. Bapak Dayat selaku Supervisor disalah satu Operator Telekomunikasi Seluler cabang Batam yang telah memberi bantuan penjelasan untuk mendukung dan menunjang penelitian proyek akhir ini.
4. Teman-teman sekalian yang selalu ada untuk membantu menyelesaikan penyusunan proyek akhir ini.
5. Semua pihak yang telah membantu dalam menyelesaikan proyek akhir ini yang tidak dapat disebutkan satu per satu

Penulis menyadari bahwa selama mengerjakan Tugas Akhir banyak hal-hal yang dapat dipelajari dan semuanya tidak lepas dari kesalahan maupun kekurangan. Akhir kata mohon maaf atas kesalahan dan kekurangan serta keterbatasan pada data dan dokumentasi. Kritik saran yang membangun sangatlah diharapkan. Semoga

laporan Tugas Akhir ini dapat bermanfaat dan bisa disempurnakan pada masa yang akan datang.

Batam, 12 Januari 2017

Penulis

ABSTRAK

APLIKASI PENJUALAN PULSA ELEKTRONIK BERBASIS ANDROID

Seiring perkembangan teknologi ponsel saat ini, kebutuhan pengguna akan pulsa sebagai alat pembayaran untuk penggunaan ponsel juga semakin besar. Hal ini membuat semakin banyak bermunculan agen penjual pulsa baik secara mandiri maupun dengan membuka toko. Namun transaksi penjualan pulsa yang ada masih dilakukan secara manual dan justru dianggap kurang efisien dan merepotkan. Karena itu dibutuhkan suatu aplikasi yang efisien dan mudah untuk membantu para agen pulsa dalam melakukan transaksi penjualan pulsa.

Aplikasi penjualan pulsa elektronik ini merupakan aplikasi yang dirancang sedemikian rupa sehingga *user-friendly* dan dapat mempermudah penjual pulsa dalam melakukan transaksi. Aplikasi ini dapat digunakan pada *smartphone* berbasis android dengan minimal versi android Kitkat.

Manfaat dari aplikasi ini adalah untuk memudahkan dan mempercepat transaksi pengiriman pulsa oleh penjual. Aplikasi ini diharapkan mampu mempermudah penjual pulsa dalam melakukan transaksi dengan menyediakan laporan riwayat transaksi, menandai pelanggan yang belum melunasi pembayaran, dan laporan pendapatan penjual yang valid serta menyediakan format laporan transaksi secara otomatis setelah pengguna melakukan beberapa pengaturan di awal penggunaan aplikasi.

Kata kunci : aplikasi, pulsa, penjualan, transaksi, Android.

ABSTRACT

APPLICATION OF ELECTRONIC PULSE SALES BASED ON ANDROID

As the development of the current mobile phone technology, user will needs pulse as payment for the use of mobile phones are also getting bigger. This makes it more and more popping up dealer pulse either independently or by opening a store. But sales of existing toll transactions are still done manually and thus considered less efficient and hassle. Because it needed an efficient and easy application to assist the Agency in performing the transaction of sale pulse.

Application of electronic pulse sales is an application with user-friendly design and can make the seller a pulse easier in the transaction process. This application can be used on android smartphone with minimal version android kitkat.

The benefit of this application is to facilitate and accelerate the transaction delivery of pulses by the seller. This application is expected to facilitate the seller pulse doing transaction by providing a transaction history, mark the customer's debt and reports a valid seller's income as well as provide a report format the transaction automatically after the user do some settings at the beginning of the use of the application.

Key words: application, pulse, sales, transaction, Android.

Daftar Isi

ABSTRAK	vi
Daftar Isi.....	viii
Daftar Gambar.....	xii
Daftar Tabel	xv
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Tinjauan Pustaka.....	3
1.7 Sistematika Penulisan	5
Bab I : Pendahuluan	5
Bab II : Landasan Teori.....	5
Bab III : Analisis dan Perancangan.....	5
Bab IV : Implementasi dan Pengujian	5
BAB II.....	6
LANDASAN TEORI.....	6
2.1 Short Message Service (SMS)	6
2.2 Unified Modeling Language (UML)	8
2.2.1 Use case diagram.....	8
2.2.2 Activity diagram.....	9
2.2.3 Sequence diagram	9
2.2.4 Class diagram	9
2.3 Android	10
2.4 Basis Data (<i>Database</i>)	11
2.4.1 Definisi basis data	11

2.4.2 SQLite	12
2.4.3 Structured Query Language (SQL)	12
2.4.4 Entity Relationship Diagram (ERD)	12
2.4.5 Physical Data Modelling (PDM).....	12
2.5 Android Studio.....	13
BAB III.	14
ANALISIS DAN PERANCANGAN.....	14
3.1 Deskripsi Umum Sistem	14
3.2 Kebutuhan Fungsional Sistem	15
3.3 Kebutuhan Nonfungsional Sistem	16
3.4 Use Case Diagram	17
3.5 Skenario <i>Use Case</i>	17
3.6 Activity Diagram	24
3.6.1 <i>Activity diagram</i> login dengan kata sandi	24
3.6.2 <i>Activity diagram</i> login dengan PIN transaksi.....	25
3.6.3 <i>Activity diagram</i> mengganti kata sandi aplikasi.....	26
3.6.4 <i>Activity diagram</i> menyimpan data operator	26
3.6.5 <i>Activity diagram</i> memperbaharui data operator.....	27
3.6.6 <i>Activity diagram</i> mengatur nomor SMS center.....	28
3.6.7 <i>Activity diagram</i> menyimpan PIN transaksi	29
3.6.8 <i>Activity diagram</i> mengatur karakter pemisah	29
3.6.9 <i>Activity diagram</i> mengganti pin transaksi.....	30
3.6.10 <i>Activity diagram request</i> pengisian pulsa	31
3.6.11 <i>Activity diagram</i> melihat riwayat transaksi.....	32
3.6.12 <i>Activity diagram</i> menghapus riwayat transaksi.....	32
3.6.13 <i>Activity diagram</i> cek laporan hutang.....	33
3.6.14 <i>Activity diagram</i> mengganti status pembayaran	33
3.6.15 <i>Activity diagram</i> lihat pesan masuk	34
3.6.16 <i>Activity diagram</i> cek saldo	34
3.7 Sequence Diagram	35
3.7.1 Sequence diagram login	35

3.7.2	<i>Sequence diagram</i> mengganti kata sandi	35
3.7.3	<i>Sequence diagram</i> memperbaharui data operator	36
3.7.4	<i>Sequence diagram</i> mengatur nomor SMS center.....	36
3.7.5	<i>Sequence diagram</i> menghapus sms center	37
3.7.6	<i>Sequence diagram</i> menyimpan karakter pemisah.....	37
3.7.7	<i>Sequence diagram</i> mengganti PIN transaksi	38
3.7.8	<i>Sequenece diagram</i> request pengisian pulsa.....	38
3.7.9	<i>Sequence diagram</i> melihat riwayat transaksi.....	39
3.7.10	<i>Sequence diagram</i> menghapus riwayat transaksi.....	39
3.7.11	<i>Sequence diagram</i> melihat laporan hutang	40
3.7.12	<i>Sequence diagram</i> mengganti status pembayaran	40
3.7.13	<i>Sequence Diagram</i> lihat pesan masuk	40
3.7.16	<i>Sequence Diagram</i> cek saldo	41
3.8	Class Diagram.....	42
3.9	Perancangan Basis Data.....	43
3.9.1	Entity Relationship Diagram (ERD)	43
3.9.2	Physical Data Modelling (PDM).....	44
3.10	Perancangan Antarmuka	45
3.10.1	Perancangan antarmuka login	45
3.10.2	Perancangan antarmuka menu utama.....	47
3.10.3	Perancangan antarmuka transaksi	48
3.10.4	Perancangan antarmuka riwayat transaksi	49
3.10.5	Perancangan antarmuka menu pengaturan.....	50
3.10.6	Perancangan antarmuka menu utama pengaturan operator.....	51
3.10.7	Perancangan antarmuka pengaturan SMS center	53
3.10.8	Perancangan antarmuka pengaturan karakter pemisah	54
3.10.9	Perancangan antarmuka pengaturan PIN transaksi	55
3.10.11	Perancangan antarmuka pengaturan kata sandi.....	56
3.10.12	Perancangan antarmuka pesan masuk.....	57
3.10.13	Perancangan antarmuka cek saldo	58
3.10.14	Perancangan antarmuka data hutang.....	59

3.11 Perancangan Algoritma.....	59
Perhitungan Total Pendapatan Penjual Pulsa.....	59
BAB IV.....	61
IMPLEMENTASI DAN PENGUJIAN	61
4.1 Implementasi Struktur Tabel Basis Data	61
4.2 Implementasi Proses	63
4.2.1 Implementasi proses login	63
4.2.2 Implementasi proses transaksi pengiriman pulsa.....	65
4.2.3 Implementasi proses pengelolaan riwayat transaksi	66
4.2.4 Implementasi proses <i>request</i> cek saldo	68
4.2.5 Implementasi proses pengecekan inbox.....	70
4.2.6 Implementasi proses pengelolaan data hutang.....	71
4.2.7 Implementasi proses pengaturan operator.....	73
4.2.8 Implementasi proses pengaturan nomor SMS center.....	74
4.2.9 Implementasi proses pengaturan karakter pemisah.....	76
4.2.10 Implementasi proses pengaturan PIN transaksi	77
4.2.11 Implementasi proses mengubah kata sandi aplikasi	78
4.3 Pengujian	80
BAB V.....	86
KESIMPULAN DAN SARAN.....	86
5.1 Kesimpulan	86
5.2 Saran	86
DAFTAR PUSTAKA	87

Daftar Gambar

Gambar 2. 1	Arsitektur SMS yang terintegrasi dengan jaringan GSM.....	6
Gambar 2. 2	Diagram alir SMS.....	7
Gambar 3. 1	Deskripsi umum sistem	14
Gambar 3. 3	<i>Activity diagram</i> login dengan kata sandi	24
Gambar 3. 5	<i>Activity diagram</i> mengganti kata sandi aplikasi.....	26
Gambar 3. 6	<i>Activity diagram</i> memperbaharui data operator	26
Gambar 3. 7	<i>Activity diagram</i> memperbaharui data operator	27
Gambar 3. 8	<i>Activity diagram</i> mengatur nomor SMS center.....	28
Gambar 3. 9	<i>Activity diagram</i> menyimpan PIN transaksi.....	29
Gambar 3. 10	<i>Activity diagram</i> mengatur karakter pemisah.....	29
Gambar 3. 11	<i>Activity diagram</i> mengganti PIN transaksi.....	30
Gambar 3. 12	<i>Activity diagram</i> request pengisian pulsa.....	31
Gambar 3. 13	<i>Activity diagram</i> melihat riwayat transaksi	32
Gambar 3. 14	<i>Activity diagram</i> menghapus riwayat transaksi.....	32
Gambar 3. 15	<i>Activity diagram</i> cek laporan hutang.....	33
Gambar 3. 16	<i>Activity diagram</i> mengganti status pembayaran.....	33
Gambar 3. 17	<i>Activity diagram</i> use case lihat pesan masuk	34
Gambar 3. 18	<i>Activity diagram</i> use case cek saldo	34
Gambar 3. 19	<i>Sequence diagram</i> login	35
Gambar 3. 20	<i>Sequence diagram</i> mengganti kata sandi	35
Gambar 3. 21	<i>Sequence diagram</i> memperbaharui data operator	36
Gambar 3. 22	<i>Sequence diagram</i> mengatur nomor SMS center	36
Gambar 3. 23	<i>Sequence diagram</i> menghapus SMS center	37
Gambar 3. 24	<i>Sequence diagram</i> mengatur karakter pemisah.....	37
Gambar 3. 25	<i>Sequence diagram</i> mengganti PIN transaksi.....	38
Gambar 3. 26	<i>Sequence diagram</i> request pengisian pulsa.....	38
Gambar 3. 27	<i>Sequence diagram</i> melihat riwayat transaksi	39
Gambar 3. 28	<i>Sequence diagram</i> menghapus riwayat transaksi.....	39
Gambar 3. 29	<i>Sequence diagram</i> melihat laporan hutang	40
Gambar 3. 30	<i>Sequence diagram</i> mengganti status pembayaran.....	40

Gambar 3. 31 <i>Sequence diagram</i> lihat pesan masuk.....	41
Gambar 3. 32 <i>Sequence diagram</i> cek saldo	41
Gambar 3. 33 <i>Class diagram</i> aplikasi	42
Gambar 3. 34 <i>Entity relationship diagram</i>	43
Gambar 3. 35 <i>Physical data modelling</i>	44
Gambar 3. 36 Perancangan antarmuka login dengan kata sandi aplikasi	45
Gambar 3. 37 Perancangan antarmuka login dengan PIN transaksi	46
Gambar 3. 38 Perancangan antarmuka menu utama.....	47
Gambar 3. 39 Perancangan antarmuka transaksi	48
Gambar 3. 40 Perancangan antarmuka riwayat transaksi	49
Gambar 3. 41 Perancangan antarmuka menu pengaturan.....	50
Gambar 3. 42 Perancangan antarmuka menu utama pengaturan operator.....	51
Gambar 3. 43 Perancangan antarmuka pengaturan operator	52
Gambar 3. 44 Perancangan antarmuka pengaturan SMS Center	53
Gambar 3. 45 Perancangan antarmuka pengaturan karakter pemisah	54
Gambar 3. 46 Perancangan antarmuka pengaturan PIN transaksi	55
Gambar 3. 47 Perancangan antarmuka pengaturan kata sandi.....	56
Gambar 3. 48 Perancangan antarmuka pesan masuk.....	57
Gambar 3. 49 Perancangan antarmuka cek saldo.....	58
Gambar 3. 50 Perancangan antarmuka data hutang	59
Gambar 4. 1 Implementasi antarmuka login dengan kata sandi	63
Gambar 4. 2 Implementasi antarmuka login dengan PIN transaksi.....	63
Gambar 4. 3 <i>Source code</i> proses login dengan kata sandi yang terdapat pada file loginKataSandi.java	64
Gambar 4. 4 <i>Source code</i> proses login dengan PIN transaksi yang terdapat pada file loginPin.java	64
Gambar 4. 5 Implementasi antarmuka transaksi	65
Gambar 4. 6 <i>Source code</i> proses transaksi pada file TransaksiFragment.java	66
Gambar 4. 7 Implementasi antarmuka riwayat transaksi	66
Gambar 4. 8 <i>Source code</i> proses pengelolaan riwayat transaksi pada file ControllerRiwayat.java	68

Gambar 4. 9 Implementasi antarmuka cek saldo	68
Gambar 4. 10 <i>Source code</i> proses <i>request</i> cek saldo pada file CekSaldoFragment.java	70
Gambar 4. 11 Implementasi antarmuka inbox	70
Gambar 4. 12 <i>Source code</i> proses <i>pengecekan inbox</i> pada file InboxFragment.java	71
Gambar 4. 13 Implementasi antarmuka data hutang.....	71
Gambar 4. 14 <i>Source code</i> proses pengelolaan data hutang pada file ControllerHutang.java.....	72
Gambar 4. 15 Implementasi antarmuka pengaturan operator	73
Gambar 4. 16 <i>Source code</i> proses pengaturan operator pada file ControllerOperator.java	74
Gambar 4. 17 Implementasi antarmuka pengaturan SMS center.....	74
Gambar 4. 18 <i>Source code</i> proses pengaturan SMS center pada file ControllerSMSCenter.java	75
Gambar 4. 19 Implementasi antarmuka pengaturan karakter pemisah	76
Gambar 4. 20 <i>Source code</i> proses pengaturan karakter pemisah pada file ControllerKarakterPemisah.java	77
Gambar 4. 21 Implementasi antarmuka pengaturan PIN transaksi.....	77
Gambar 4. 22 <i>Source code</i> proses pengaturan PIN transaksi pada file ControllerPengaturanPin.java	78
Gambar 4. 23 Implementasi antarmuka pengaturan kata sandi aplikasi	78
Gambar 4. 24 <i>Source code</i> proses mengubah kata sandi aplikasi pada file ControllerPengaturanKataSandi.java	79

Daftar Tabel

Tabel 1. 1 Perbandingan penelitian.....	4
Tabel 3. 1 Skenario <i>use case</i> login dengan kata sandi	17
Tabel 3. 2 Skenario <i>use case</i> login dengan PIN transaksi.....	18
Tabel 3. 3 Skenario <i>use case</i> mengganti kata sandi aplikasi.....	18
Tabel 3. 4 Skenario <i>use case</i> menyimpan data operator	19
Tabel 3. 5 Skenario <i>use case</i> memperbaharui data operator	19
Tabel 3. 6 Skenario <i>use case</i> mengatur nomor sms center	20
Tabel 3. 7 Skenario <i>use case</i> menyimpan pin transaksi.....	20
Tabel 3. 8 Skenario <i>use case</i> mengatur karakter pemisah	21
Tabel 3. 9 Skenario <i>use case</i> mengganti pin transaksi.....	21
Tabel 3. 10 Skenario <i>use case request</i> pengisian pulsa	21
Tabel 3. 11 Skenario <i>use case</i> melihat riwayat transaksi.....	22
Tabel 3. 12 Skenario <i>use case</i> menghapus riwayat transaksi.....	22
Tabel 3. 13 Skenario <i>use case</i> melihat laporan hutang	23
Tabel 3. 14 Skenario <i>use case</i> mengganti status pembayaran	23
Tabel 3. 15 Skenario <i>use case</i> lihat pesan masuk	23
Tabel 3. 16 Skenario <i>use case</i> cek saldo	24
Tabel 3. 17 Deskripsi perancangan antarmuka login dengan kata sandi aplikasi .	45
Tabel 3. 18 Deskripsi perancangan antarmuka login dengan PIN transaksi.....	46
Tabel 3. 19 Deskripsi perancangan antarmuka menu utama.....	47
Tabel 3. 20 Deskripsi perancangan antarmuka transaksi	48
Tabel 3. 21 Deskripsi perancangan antarmuka riwayat transaksi	49
Tabel 3. 22 Deskripsi perancangan antarmuka menu pengaturan.....	50
Tabel 3. 23 Deskripsi perancangan antarmuka menu utama pengaturan operator	51
Tabel 3. 24 Deskripsi perancangan antarmuka pengaturan operator	52
Tabel 3. 25 Deskripsi perancangan antarmuka pengaturan SMS Center.....	54
Tabel 3. 26 Deskripsi perancangan antarmuka pengaturan karakter pemisah	55
Tabel 3. 27 Deskripsi perancangan antarmuka pengaturan PIN transaksi.....	55
Tabel 3. 28 Perancangan antarmuka pengaturan kata sandi	56
Tabel 3. 29 Deskripsi perancangan antarmuka pesan masuk.....	57

Tabel 3. 30 Deskripsi perancangan antarmuka cek saldo	58
Tabel 3. 31 Deskripsi perancangan antarmuka data hutang.....	59
Tabel 4. 1 Implementasi struktur tabel basis data operator.....	61
Tabel 4. 2 Implementasi struktur tabel basis data transaksi.....	61
Tabel 4. 3 Implementasi struktur tabel basis data smscenter	61
Tabel 4. 4 Implementasi penyimpanan di dalam file teks.....	62
Tabel 4. 5 Tabel file sumber proses login	64
Tabel 4. 6 Tabel file sumber proses transaksi	65
Tabel 4. 7 Tabel file sumber proses riwayat transaksi	67
Tabel 4. 8 Tabel file sumber proses <i>request</i> cek saldo	69
Tabel 4. 9 Tabel file sumber proses pengecekan inbox	71
Tabel 4. 10 Tabel file sumber proses pengelolaan data hutang	72
Tabel 4. 11 Tabel file sumber proses pengaturan operator	73
Tabel 4. 12 Tabel file sumber proses pengaturan nomor SMS center	75
Tabel 4. 13 Tabel file sumber proses pengaturan karakter pemisah.....	76
Tabel 4. 14 Tabel file sumber proses pengaturan PIN transaksi.....	77
Tabel 4. 15 Tabel file sumber proses pengaturan kata sandi aplikasi.....	78
Tabel 4. 16 Pengujian.....	80

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teknologi ponsel telah mengalami perkembangan yang sangat pesat dan digunakan oleh berbagai kalangan di Indonesia maupun mancanegara. Bukan lagi hanya sekedar alat komunikasi, teknologi ponsel sekarang bahkan memungkinkan pengguna untuk mengkonsumsi berita, bersosialisasi di dunia maya, dan melakukan transaksi jual-beli. Ponsel saat ini didukung oleh sistem operasi *mobile* dan sudah hampir setara teknologinya jika dibandingkan dengan *Personal Computer (PC)* sehingga ponsel pada saat ini disebut juga dengan istilah *Smartphone*. Sistem operasi pada ponsel yang paling banyak digunakan saat ini adalah Android. Ponsel membutuhkan pulsa yang digunakan sebagai alat pembayaran ke penyedia jasa telekomunikasi sehingga ponsel dapat digunakan sebagaimana mestinya untuk bertukar pesan singkat (SMS), menelepon, berinternet, dan sebagainya. Dengan semakin meningkatnya kebutuhan dan permintaan pasar pada penggunaan ponsel, maka semakin tinggi pula kebutuhan dan permintaan pulsa yang menyebabkan mulai banyaknya agen penjual pulsa.

Mayoritas dari para penjual pulsa elektronik, khususnya yang memiliki kiosnya sendiri, masih melakukan transaksi dengan menggunakan sistem manual secara tulis menulis untuk merekap data pelanggannya kebuku, terpisah dengan ponsel yang berfungsi untuk mengirimkan pesan permintaan pengisian pulsa dalam bentuk SMS ke server penyedia pulsa. Hal ini tentunya menyulitkan bagi para penjual pulsa dalam menghitung pendapatan dari hasil penjualan pulsa. Ditambah lagi jika penjual lupa bahwa ada pelanggan yang belum melunasi pembayaran (hutang) maka akan merugikan bagi si penjual. Permasalahan lain yang penulis temukan dalam transaksi penjualan pulsa adalah penjual kesulitan mengingat format transaksi kode pengisian pulsa yang sesuai dengan yang telah ditentukan oleh server penyedia pulsa. Penjual akan selalu mengetik ulang seluruh format transaksi kode pengisian pulsa setiap kali ada transaksi baru. Jika penjual salah mengisi format

transaksi kode pengisian pulsa, mengakibatkan pulsa tidak akan masuk kenomor tujuan atau terkadang bisa masuk kenomor tujuan yang salah. Berdasarkan permasalahan inilah penulis ingin meminimalisir kesalahan dalam proses transaksi sekaligus mempermudah pengisian pulsa pelanggan oleh penjual dengan membuat “Aplikasi Penjualan Pulsa Elektronik Berbasis Android”.

Aplikasi yang akan penulis bangun ini diharapkan akan mempermudah para penjual pulsa elektronik. Dengan adanya aplikasi ini, para penjual pulsa bisa mengetahui pendapatan dari hasil penjualan pulasanya dalam periode waktu tertentu dan mengetahui pelanggan mana saja yang belum melunasi pembayaran (hutang). Aplikasi ini juga dapat menampilkan riwayat pengiriman pulsa dan menerima SMS laporan status transaksi telah sukses atau gagal dari server penyedia pulsa untuk mencegah penipuan dalam proses transaksi. Aplikasi akan menyimpan format pengiriman pulsa untuk setiap operator sehingga penjual pulsa tidak perlu mengetikkan ulang setiap kali ingin melakukan transaksi pengisian pulsa pelanggan.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan sebelumnya, maka rumusan masalah pada aplikasi penjualan pulsa elektronik ini adalah sebagai berikut :

1. Bagaimana membangun aplikasi Android yang dapat melakukan permintaan pengisian pulsa pelanggan ke server penyedia pulsa tanpa mengharuskan penjual menghafal format kode transaksi pengisian pulsa?
2. Bagaimana aplikasi Android yang dibangun dapat mengelola data penjualan pulsa?
3. Bagaimana aplikasi Android yang dibangun dapat mencatat hutang yang dilakukan pelanggan saat transaksi pembelian pulsa?

1.3 Batasan Masalah

Batasan masalah dalam pembuatan aplikasi ini adalah sebagai berikut :

1. Aplikasi tidak mendukung untuk permintaan perubahan pin transaksi penjual pulsa.
2. Aplikasi tidak menyediakan fitur penambahan saldo/stok pulsa.

3. Aplikasi tidak bisa digunakan untuk permintaan pengisian pulsa beberapa nomor dalam satu transaksi.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah merancang dan membangun aplikasi pada ponsel dengan sistem operasi Android yang digunakan untuk melakukan transaksi pengisian pulsa elektronik, merekap data penjualan pulsa, serta mengelola data hutang pelanggan yang belum membayar pulsa yang dibelinya.

1.5 Manfaat Penelitian

Dengan terselesaikannya tugas akhir ini, diharapkan beberapa manfaat yang akan diperoleh adalah antara lain :

1. Memberi kemudahan bagi agen penjual pulsa dalam melakukan transaksi pengisian pulsa elektronik.
2. Meminimalisir kesalahan dalam pengisian format transaksi kode pengisian pulsa.
3. Aplikasi mampu merekap data penjualan pulsa dan meminimalisir kerugian penjual pulsa.
4. Bagi mahasiswa-mahasiswa lainnya agar dapat digunakan sebagai bahan studi banding untuk penelitian selanjutnya.

1.6 Tinjauan Pustaka

Laporan ini merupakan pengembangan dari laporan proyek akhir II yang dibuat oleh Purba dkk (2015) di Politeknik Negeri Batam yang berjudul “Aplikasi Penjualan Pulsa Elektronik”. Pada laporan sebelumnya, aplikasi ini dibangun untuk digunakan sebagai aplikasi berbasis *desktop* dengan dibantu oleh modem GSM sebagai sarana mengirimkan SMS permintaan pengisian pulsa ke server penyedia pulsa. Hasil akhir yang telah didapatkan adalah aplikasi berhasil mengelola data transaksi pengisian pulsa pelanggan dengan menyediakan laporan pendapatan, laporan transaksi, serta laporan hutang.

Penelitian serupa juga pernah dilakukan oleh Yulianti dan Rusyiatno (2013) dengan membangun aplikasi berbasis SMS untuk transaksi pembelian pulsa dan

penjualan voucher pulsa menggunakan Visual Basic. Fitur-fitur yang dikembangkan adalah mengelola data *supplier* pulsa, data pembelian pulsa beserta saldo awal dan akhir, data penjualan voucher pulsa, serta melakukan transaksi pengisian pulsa pelanggan dan penjualan voucher pulsa. Kelebihan dari sistem yang dibangun adalah lebih hemat biaya dalam pengecekan saldo penjual.

Penelitian lainnya dilakukan oleh Setya (2015), mahasiswa Universitas Muhammadiyah Sidoarjo yang berfokus pada pengembangan sistem aplikasi penjualan pulsa elektronik berbasis Android. Hasil yang didapatkan adalah aplikasi mampu membantu penjual pulsa untuk melakukan transaksi dengan lebih efektif dan efisien karena penjual pulsa tidak perlu mengetikkan ulang format transaksi kode pengisian pulsa setiap melakukan transaksi. Penjual hanya perlu memilih menu-menu yang tersedia pada aplikasi. Tetapi aplikasi yang dibangun belum menyediakan pengelolaan data hutang pelanggan.

Perbedaan dari ketiga acuan yang telah dijabarkan sebelumnya terhadap rancangan dan pembangunan aplikasi penjualan pulsa elektronik berbasis Android yang penulis susun adalah aplikasi ini dikembangkan dengan menggunakan editor Android Studio. Aplikasi ini menyediakan fitur transaksi pengiriman pulsa, penyimpanan riwayat transaksi, perhitungan total pendapatan, merekap SMS yang masuk dari server pulsa, dan menyediakan fitur untuk menandai pelanggan yang belum melunasi pembayaran. Aplikasi ini bisa digunakan untuk semua jenis kartu GSM yang telah terdaftar sebagai agen penjual pulsa. Bisa melakukan transaksi dengan format transaksi menggunakan kode prefix maupun tidak menggunakan kode prefix.

Tabel 1. 1 Perbandingan penelitian

Penulis	Yulianti dan Rusyiatno (2013)	Setya (2015)	Purba, dkk (2015)	Tugas Akhir ini
Judul Penelitian	Rekayasa Perangkat Lunak Penjualan Pulsa Electric Berbasis Sms Menggunakan Visual Basic 6.0	Aplikasi Penjualan Pulsa Elektronik Berbasis Android	Aplikasi Penjualan Pulsa Elektronik	Aplikasi Penjualan Pulsa Elektronik Berbasis Android
Software	Visual Basic	Android Studio, Emulator Android	Netbeans IDE 8.1, xampp, Gammu	Android Studio, Emulator Android
Platform aplikasi	<i>Desktop</i>	<i>Mobile</i> berbasis Android	<i>Desktop</i>	<i>Mobile</i> berbasis Android

Penulis	Yulianti dan Rusyiatno (2013)	Setya (2015)	Purba, dkk (2015)	Tugas Akhir ini
Fitur-fitur	<ul style="list-style-type: none"> Mengelola data, pembuatan laporan, serta monitoring data pembelian dan penjualan pulsa. Menghemat biaya dalam proses pengecekan saldo 	<ul style="list-style-type: none"> Pengelolaan data riwayat transaksi Pengelolaan data jumlah transaksi 	<ul style="list-style-type: none"> Merekap riwayat transaksi Mengelola laporan pendapatan Menyediakan laporan hutang 	<ul style="list-style-type: none"> Merekap riwayat transaksi Mengelola laporan pendapatan Menyediakan laporan hutang Menyediakan fitur untuk pengecekan pesan masuk

1.7 Sistematika Penulisan

Agar lebih memudahkan dalam penyusunan laporan penelitian, maka penulis membuat sistematika penulisan laporan sebagai berikut :

Bab I : Pendahuluan

Bab ini berisi tentang alasan penulis memilih judul atau latar belakang masalah, perumusan masalah, pembatasan masalah, tujuan penelitian, tinjauan pustaka, dan sistematika penulisan.

Bab II : Landasan Teori

Bab ini berisi tentang teori-teori dari berbagai referensi yang berkaitan dengan aplikasi penjualan pulsa elektronik berbasis Android yang mendasari dan menunjang penelitian.

Bab III : Analisis dan Perancangan

Bab ini berisi tentang analisis perancangan aplikasi penjualan pulsa elektronik. Mulai dari deskripsi umum sistem, kebutuhan fungsional sistem, kebutuhan non-fungsional sistem, diagram *use case*, skenario *use case*, *activity diagram*, *sequence diagram*, *class diagram*, perancangan basis data, dan perancangan antarmuka.

Bab IV : Implementasi dan Pengujian

Bab ini berisi tentang implementasi dan pengujian aplikasi yang sebelumnya telah dirancang. Apakah sudah sesuai dengan kebutuhan dan layanan yang diharapkan.

Bab V : Kesimpulan dan Saran

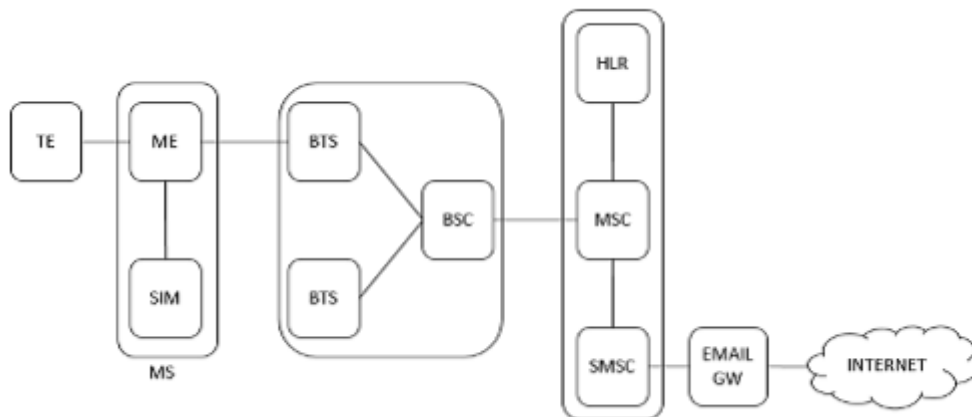
Bab ini berisi tentang kesimpulan dari rangkaian penelitian yang telah dilaksanakan dan saran penulis untuk penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Short Message Service (SMS)

Menurut Gunawan dkk (2008), *Short Message Service (SMS)* merupakan salah satu fasilitas dari teknologi *Global System for Communication (GSM)* yang memungkinkan mengirim dan menerima pesan-pesan singkat berupa teks dengan kapasitas maksimal 160 karakter dari *Mobile Station (MS)*. Kapasitas maksimal ini tergantung dari alphabet yang digunakan, untuk alphabet latin maksimal 160 karakter, dan untuk non latin seperti alphabet arab atau china maksimal 70 karakter.

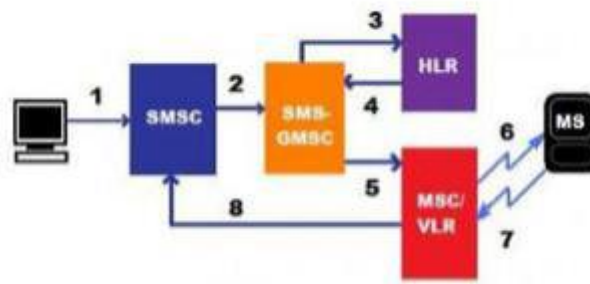


Gambar 2. 1 Arsitektur SMS yang terintegrasi dengan jaringan GSM

Berdasarkan Gambar 2.1 menunjukkan arsitektur dari SMS yang di integrasikan dengan jaringan GSM. Ada tiga elemen tambahan yaitu *short message entity (SME)*, *Short Message Service Center (SMSC)* dan *email gateway*.

- a) *Terminal Equipment (TE)* merupakan perangkat yang di gunakan seperti hp yang terhubung dengan ME
- b) *Mobile Equipment (ME)* terdiri dari pemancar radio, display dan DSP
- c) *Subscribe Identity Module (SIM)* berfungsi untuk mengidentifikasi user pada jaringan
- d) *Base Transceiver Station (BTS)* terdiri dari pemancar radio untuk berkomunikasi dengan MS

- e) *Base Station Controller (BSC)* berfungsi untuk mengatur radio *resources* untuk satu atau lebih BTS
- f) *Home Location Register (HLR)* merupakan basis data yang memiliki data pelanggan tetap
- g) *Mobile Switching Center (MSC)* melaksanakan fungsi seperti registrasi, *authentication*, update lokasi.
- h) *SMS Center (SMSC)* mengatur proses pengiriman dan penerimaan pesan dari dan atau menuju SME sesuai dengan proses *store and forward*
- i) *Email gateway*, sebuah gateway yang menghubungkan antara email dan sms pada internet.



Gambar 2. 2 Diagram alir SMS

Diagram alir SMS mobile sebagai berikut

1. ME mengirim pesan ke SMSC
2. SMSC mengirim pesan ke SMS-GMSC
3. SMS-GMSC menghubungi HLR untuk informasi routing
4. HLR membalas informasi routing ke SMS-GMSC
5. SMS-GMSC meneruskan pesan ke MSC/VLR
6. MS di-paging dan koneksi terbentuk antara MS dan network, sebagaimana dalam setup panggilan normal. (dengan demikian posisi MS di ketahui dan apakah MS boleh berada dalam network/proses otentikasi)
7. Jika otentikasi berhasil, MSC/VLR mengirim pesan tersebut ke MS
8. Jika pengiriman berhasil, delivery report di kirim dari MSC/VLR ke SMSC. Namun jika tidak, MSC/VLR akan menginformasikan ke HLR dan *failure report*

di kirim ke SMS-C. pada kasus pengiriman yang gagal HLR dan VLR akan mendapat informasi “Message Waiting” yang menunjukkan ada pesan di SMSC yang menunggu untuk di kirimkan ke MS. Informasi di HLR terdiri dari list SMSC pengiriman pesan, sedangkan di VLR terdapat “flag” yang menunjukkan apakah list pesan dalam keadaan kosong atau tidak. Jika MS available dan siap menerima pesan maka,HLR akan memberi tahu SMSC.

2.2 Unified Modeling Language (UML)

UML merupakan Bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Secara fisik, UML adalah sekumpulan spesifikasi yang dikeluarkan oleh *Object Management Group* (OMG), sebuah konsorsium terbuka yang terdiri dari banyak perusahaan (Rosa dan Shalahuddin, 2011). UML biasa digunakan untuk (Henderi, 2008):

- a. Menggambarkan batasan sistem dan fungsi-fungsi sistem secara umum, dibuat dengan *use case* dan *aktor*.
- b. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams*.
- c. Menggambarkan representasi struktur statik sebuah sistem dalam bentuk *class diagram*.
- d. Membuat model *behaviour* yang menggambarkan kebiasaan atau sifat sebuah sistem dengan *state transition diagram*.
- e. Menyatakan arsitektur implementasi fisik menggunakan *component* dan *deployment diagram*.

2.2.1 Use case diagram

Use case diagram merupakan permodelan untuk kelakuan (*behaviour*) sistem yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada 2 hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case* (Rosa dan Shalahuddin, 2011):

- a) Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem yang akan dibuat. Jadi meskipun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang.
- b) *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antarunit atau aktor.

2.2.2 Activity diagram

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis. *Activity diagram* banyak digunakan mendefinisikan hal-hal berikut (Rosa dan Shalahuddin, 2011):

- a) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
- b) Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
- c) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya

2.2.3 Sequence diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek (Rosa dan Shalahuddin, 2011). *Sequence diagram* sangat bagus untuk memperlihatkan kolaborasi antar objek dan menampilkan *behaviour* beberapa objek di dalam *use case* tunggal (Fowler, 2005).

2.2.4 Class diagram

Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. *Class diagram* juga menunjukkan properti dan operasi sebuah kelas dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut (Fowler, 2005).

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut (Rosa dan Shalahuddin, 2011):

- a) Kelas main: kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
- b) Kelas yang menangani tampilan sistem: kelas yang mendefinisikan dan mengatur tampilan ke pemakai.
- c) Kelas yang diambil dari pendefinisian *use case*: kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*.
- d) Kelas yang diambil dari pendefinisian data: kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

2.3 Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi yang diinginkan. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel/*smartphone*. Kemudian untuk mengembangkan Android dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan piranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia (Safaat, 2012).

Sistem operasi Android menyediakan berbagai lokasi yang bisa digunakan untuk menyimpan data secara permanen dan yang paling lazim antara lain (Hernita dkk, 2013):

- a) *Shared preferences*: penyimpanan data permanen dalam bentuk pasangan kunci
- b) *Internal storage*: merupakan lokasi dimana file disimpan di *internal storage*. Secara default, file akan disimpan di *internal storage* yang merupakan bagian dari aplikasi. Ketika pengguna meng-*uninstall* aplikasi maka data pada *internal storage* juga akan ikut terhapus.
- c) *Local cache*: data disimpan dengan kapasitas sistem yang lebih rendah

- d) *External storage*: Setiap perangkat android memiliki *external storage* yang digunakan untuk menyimpan file. Bisa berupa SD card atau yang lainnya.
- e) *Database SQLite*: merupakan database *embedded* yang bisa diandalkan pada sistem operasi Android.

Software Development Kit (SDK)

Android SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk mengembangkan aplikasi pada *platform* Android yang menggunakan bahasa pemrograman Java. Android merupakan perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi yang direlease oleh Google. Saat ini disediakan Android SDK (Software Development Kit) sebagai alat bantu dan API untuk mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Sebagai *platform* Android aplikasi netral, Android memberi Anda kesempatan untuk membuat Aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan *Handphone/Smartphone* (Safaat, 2012).

Java Development Kit (JDK)

Java adalah sebuah teknologi yang diperkenalkan oleh Sun Microsystems pada pertengahan tahun 1990. Menurut definisi Sun, Java adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer *standalone* ataupun pada lingkungan jaringan. Untuk membuat program Java dibutuhkan kompiler dan interpreter untuk program Java berbentuk Java Development Kit (JDK) yang diproduksi oleh Sun Microsystems. Sebelum memulai instalasi Android SDK, terlebih dahulu kita harus melakukan instalasi JDK di komputer. JDK yang kami gunakan untuk dapat mengompilasi aplikasi android yang kami rancang ini adalah Java SE Development Kit 7 (Safaat, 2012).

2.4 Basis Data (*Database*)

2.4.1 Definisi basis data

Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya, basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat (Rosa dan Shalahuddin, 2011).

Basis data mempunyai komponen utama, untuk lebih jelasnya berikut adalah komponen utama sistem basis data terdiri dari:

- a. Perangkat Keras (*hardware*)
- b. Sistem Operasi (*Operating system*)
- c. Basis Data (*database*)
- d. Sistem (Aplikasi/Perangkat Lunak) Pengelola Bisnis Data (DBMS)
- e. Pemakai (*user*) serta aplikasi (perangkat lunak) lain (bersifat opsional).

2.4.2 SQLite

SQLite adalah *database opensource* yang ditanam pada Android. Dukungan yang diberikan sama seperti RDBMS standard dan mendukung pula bahasa sintaks SQL. Seluruh elemen basis data (definisi data, tabel, indeks) disimpan sebagai sebuah file. Salah satu keunggulan dari SQLite adalah memakan memori yang sedikit saat dijalankan, hanya sekitar 250 Kbyte sehingga cocok untuk piranti *mobile* (Wahana Komputer, 2013).

2.4.3 Structured Query Language (SQL)

SQL adalah bahasa yang digunakan untuk mengelola data pada RDBMS. Mulai digunakan sebagai standar yang resmi pada tahun 1986 oleh ANSI (*American National Standards Institute*) dan pada tahun 1987 oleh ISO (*International Organization for Standardization*) (Rosa dan Shalahuddin, 2011).

2.4.4 Entity Relationship Diagram (ERD)

ERD merupakan pemodelan awal basis data relasional. ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika (Rosa dan Shalahuddin, 2011).

2.4.5 Physical Data Modelling (PDM)

PDM adalah model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap tabel mempunyai sejumlah kolom dimana setiap kolom memiliki nama yang unik beserta tipe datanya. PDM merupakan konsep yang menerangkan detail dari bagaimana data disimpan di dalam basis data (Rosa dan Shalahuddin, 2011).

2.5 Android Studio

Android Studio merupakan lingkungan pengembangan Android baru berdasarkan IntelliJ IDEA. Mirip dengan Eclipse dengan ADT Plugin, Android Studio menyediakan alat pengembang terintegrasi pengembangan dan debugging. Android Studio menawarkan:

1. Berbasis Gradle.
2. *Android-spesifik refactoring* dan perbaikan yang cepat.
3. Alat Lint untuk menangkap kinerja, kegunaan, versi kompatibilitas dan masalah lainnya.
4. *ProGuard* dan *app-signature*.
5. Wizard untuk design dan membuat komponen-komponen umum sebuah layout editor yang memungkinkan untuk *drag-and-drop* komponen UI, pratinjau layout pada beberapa konfigurasi layar, dan banyak lagi.
6. Built-in dukungan untuk *Google Cloud platform*, sehingga mudah untuk mengintegrasikan *Google Cloud Messaging* dan *App Engine* sebagai komponen *server-side*.

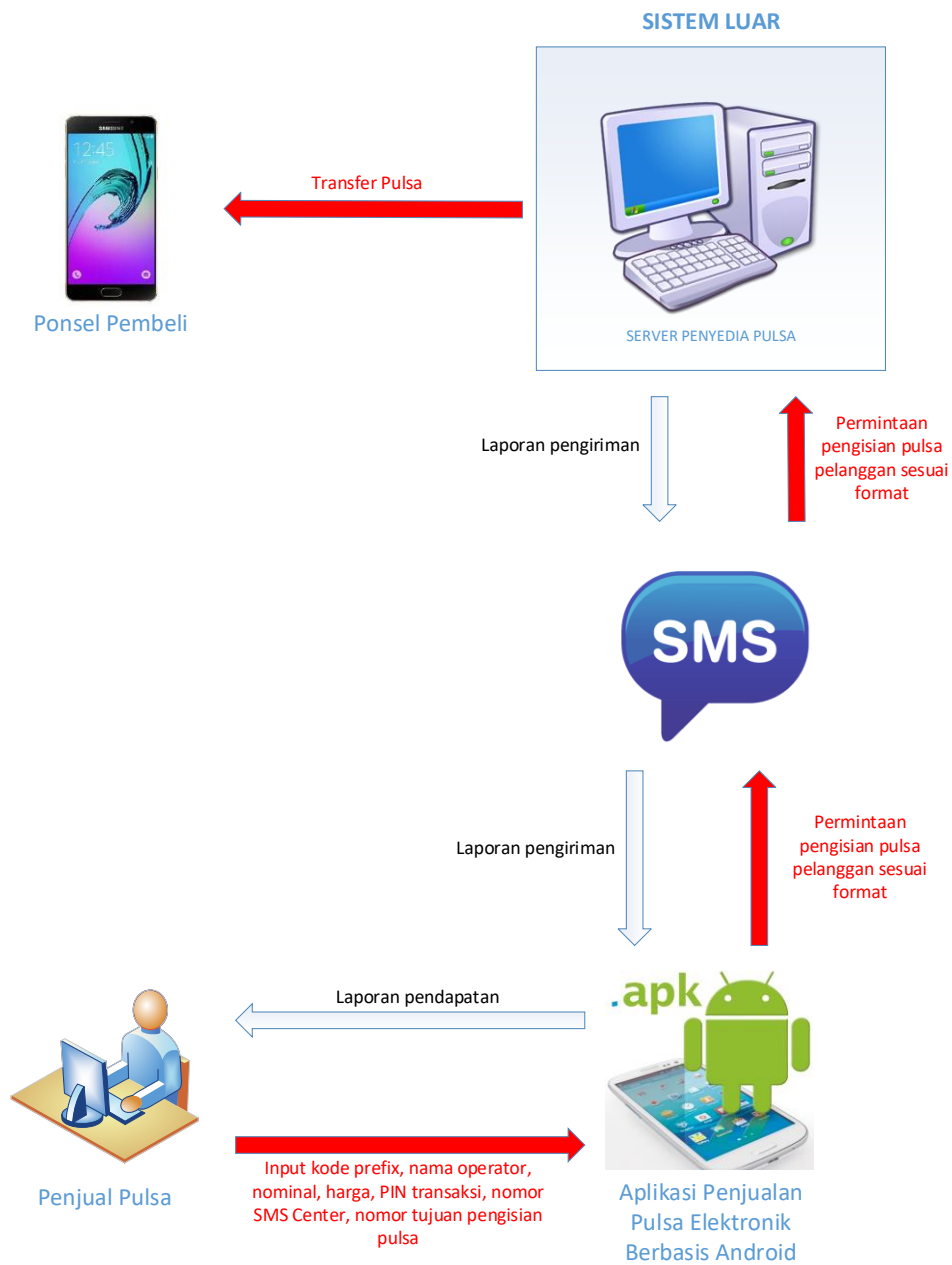
Android Studio saat ini masih dalam tahap awal pengembangan. Sebelum menginstal Android Studio perangkat komputer harus sudah terinstal Android SDK (*Software Development Kit*). SDK adalah tools API (*Application programming interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di release oleh Google.

BAB III

ANALISIS DAN PERANCANGAN

Berikut akan dijelaskan mengenai perancangan aplikasi penjualan pulsa elektronik.

3.1 Deskripsi Umum Sistem



Gambar 3. 1 Deskripsi umum sistem

Berdasarkan Gambar 3.1, aplikasi penjualan pulsa elektronik ini bekerja dengan cara mengirimkan SMS permintaan pengiriman pulsa ke server penyedia pulsa. Format transaksi kode pengisian pulsa untuk setiap operator telah tersimpan di dalam aplikasi. Penjual hanya perlu memverifikasi kode prefix masing-masing operator, kemudian aplikasi akan menyimpan kode prefix tersebut dan akan digunakan setiap kali ingin melakukan pengiriman pulsa pelanggan. Pada saat penjual ingin menggunakan aplikasi, penjual harus melakukan beberapa pengaturan terlebih dahulu untuk memperlancar proses transaksi. Yaitu melakukan pengaturan kode prefix setiap operator, pengaturan harga setiap nominal pulsa, pengaturan SMS Center sebagai nomor tujuan SMS permintaan pengiriman pulsa, dan mengatur PIN transaksi pulsa penjual yang telah terdaftar sebagai agen penjualan pulsa. Setelah melakukan transaksi, server penyedia pulsa akan mengirim SMS ke ponsel penjual yang berisi laporan pengiriman pulsa telah sukses ataupun gagal. Untuk kebutuhan perhitungan pendapatan penjualan, aplikasi akan menghitung total harga berdasarkan pengaturan harga yang sebelumnya telah diatur oleh penjual dan menghitung transaksi dimana status pembayarannya telah lunas kemudian menampilkannya selama periode waktu tertentu.

3.2 Kebutuhan Fungsional Sistem

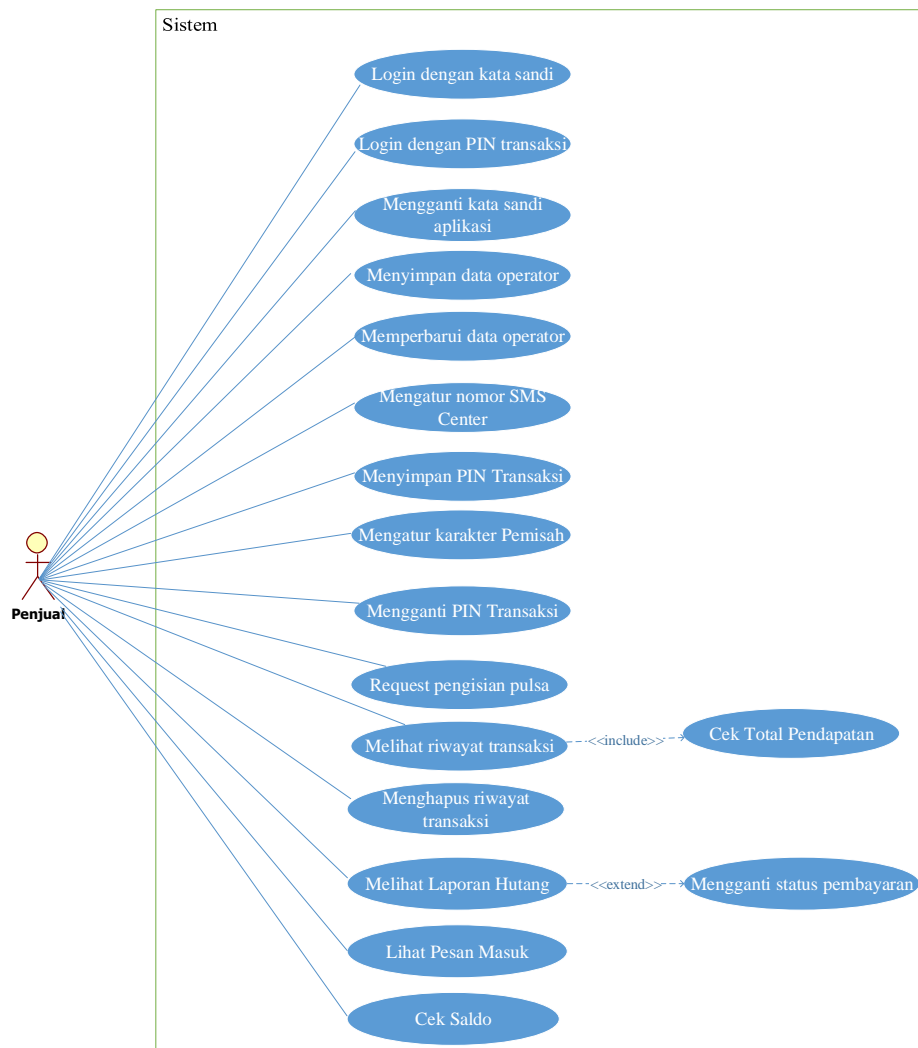
- F001 Aplikasi memiliki fitur login.
- F002 Aplikasi memiliki fitur perubahan kata sandi aplikasi.
- F003 Aplikasi dapat melakukan permintaan pengisian pulsa Telkomsel, Indosat, XL, Three, SmartFren, Flexi, Axis, Esia, Starone.
- F004 Aplikasi melakukan transaksi pulsa melalui SMS.
- F005 Aplikasi dapat menyimpan format transaksi pulsa melalui SMS untuk setiap operator.
- F006 Aplikasi dapat menyimpan riwayat transaksi penjualan pulsa.
- F007 Aplikasi dapat menghitung pendapatan penjual dari penjualan pulsa.
- F008 Aplikasi harus dapat merekam dan menyimpan rekaman waktu transaksi.
- F009 Aplikasi memiliki fitur untuk menandai pelanggan yang sudah melunaskan pembayaran atau belum.

- F010 Aplikasi menyediakan fitur untuk mengubah status pembayaran .
- F011 Aplikasi dapat menghapus data transaksi yang telah dipilih oleh penjual.
- F012 Aplikasi dapat menampilkan pesan SMS yang masuk ke kartu GSM yang digunakan penjual.
- F013 Aplikasi harus memiliki fitur untuk cek saldo.
- F014 Aplikasi dapat mengimport data nomor *handphone* dan nama dari kontak yang telah tersimpan pada ponsel penjual.
- F015 Aplikasi dapat menyimpan PIN transaksi pengiriman pulsa.
- F016 Aplikasi memiliki fitur perubahan PIN transaksi pengiriman pulsa.

3.3 Kebutuhan Nonfungsional Sistem

- NF001 Aplikasi diakses dalam bahasa Indonesia.
- NF002 Aplikasi harus mudah dioperasikan.
- NF003 Tampilan aplikasi mudah dimengerti oleh penjual.
- NF004 Aplikasi memiliki keamanan yang baik.

3.4 Use Case Diagram



Gambar 3. 2 Use case diagram

Berdasarkan Gambar 3.2, dapat dilihat diagram *use case* yang menggambarkan fungsi/layanan yang diharapkan ada pada aplikasi dan hubungannya dengan aktor. Aktor sebagai pengguna aplikasi ini adalah penjual pulsa, yang dapat melakukan 17 fungsi atau layanan yang ada pada pada aplikasi penjualan pulsa elektronik berbasis Android.

3.5 Skenario Use Case

Tabel 3. 1 Skenario use case login dengan kata sandi

Nama use case	Login dengan kata sandi
Aktor	Penjual
Deskripsi	Penjual melakukan login pada aplikasi dengan menggunakan kata sandi untuk mendapat hak akses

Kondisi Awal	Aplikasi menampilkan form login
Kondisi Akhir	Penjual mendapatkan hak akses
Skenario	1. Aplikasi mengecek apakah kata sandi sudah tersimpan atau belum
	2. Penjual memasukkan kata sandi di tempat yang sesuai
	3. Aplikasi memvalidasi kebenaran kata sandi yang telah dimasukkan penjual
	4. Aplikasi menampilkan halaman transaksi
Skenario Alternatif	1.a. [Kata sandi belum tersimpan] Lanjut ke skenario 2
	3.a. [Kata sandi salah] aplikasi menampilkan pesan kesalahan. Kembali ke skenario 2.

Tabel 3.1 mendeskripsikan alur kerja dari *use case* login dengan menggunakan kata sandi. Penjual harus memasukkan kata sandi untuk mendapatkan hak akses terhadap aplikasi.

Tabel 3. 2 Skenario *use case* login dengan PIN transaksi

Nama use case	Login dengan PIN transaksi
Aktor	Penjual
Deskripsi	Penjual melakukan login pada aplikasi dengan menggunakan PIN Transaksi untuk mendapat hak akses
Kondisi Awal	Aplikasi menampilkan form login
Kondisi Akhir	Penjual mendapatkan hak akses
Skenario	1. Penjual memilih login dengan menggunakan PIN transaksi
	2. Aplikasi mengecek apakah PIN transaksi sudah tersimpan atau belum
	3. Penjual memasukkan PIN transaksi di tempat yang sesuai
	4. Aplikasi memvalidasi kebenaran PIN transaksi yang telah dimasukkan penjual
	5. Aplikasi menampilkan halaman transaksi
Skenario Alternatif	2.a. [PIN Transaksi belum tersimpan] Lakukan skenario login dengan kata sandi
	4.a. [PIN transaksi salah] aplikasi menampilkan pesan kesalahan. Kembali ke skenario 3.

Tabel 3.2 mendeskripsikan alur kerja dari *use case* login dengan menggunakan PIN transaksi. Penjual harus memasukkan kata sandi untuk mendapatkan hak akses terhadap aplikasi.

Tabel 3. 3 Skenario *use case* mengganti kata sandi aplikasi

Nama use case	Mengganti Kata Sandi Aplikasi
Aktor	Penjual
Deskripsi	Penjual mengganti kata sandi aplikasi
Kondisi Awal	Penjual telah berhasil login ke aplikasi dan kata sandi telah terdaftar pada aplikasi
Kondisi Akhir	Kata sandi telah diganti dan disimpan
Skenario	1. Aplikasi menampilkan halaman pengaturan kata sandi dan menampilkan kata sandi yang lama
	2. Penjual memasukkan kata sandi yang baru dan konfirmasi kata sandi baru

	3. Aplikasi memvalidasi kebenaran kata sandi yang baru dan konfirmasi kata sandi baru yang telah dimasukkan
	4. Aplikasi menyimpan kata sandi yang telah diganti
Skenario Alternatif	3.a. [Kata sandi baru dan konfirmasi kata sandi baru tidak sesuai] aplikasi menampilkan pesan kesalahan. Kembali ke skenario 1.

Tabel 3.3 mendeskripsikan alur kerja dari *use case* mengganti kata sandi aplikasi.

Tabel 3. 4 Skenario *use case* menyimpan data operator

Nama use case	Menyimpan data operator
Aktor	Penjual
Deskripsi	Penjual menyimpan data operator sesuai dengan yang ditetapkan oleh server penyedia pulsa
Kondisi Awal	Data untuk setiap operator belum tersimpan pada aplikasi dan penjual telah berhasil login
Kondisi Akhir	Data untuk setiap operator telah disimpan
Skenario	1. Penjual memilih data operator yang ingin disimpan 2. Aplikasi menampilkan form pengaturan data operator 3. Penjual memasukkan kode prefix sesuai dengan operator yang dipilih 4. Penjual memasukkan harga untuk setiap nominal pembelian pulsa 5. Aplikasi memvalidasi kebenaran data yang dimasukkan penjual 6. Aplikasi menyimpan data operator
Skenario Alternatif	5.a. [Kode Prefix tidak diisi] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 3. 5.a. [harga untuk tiap nominal tidak lengkap] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 4.

Tabel 3.4 mendeskripsikan alur kerja *use case* menyimpan data operator agar tidak terjadi kesalahan pada saat melakukan transaksi pengisian pulsa pelanggan dan perhitungan total pendapatan.

Tabel 3. 5 Skenario *use case* memperbaharui data operator

Nama use case	Memperbaharui data operator
Aktor	Penjual
Deskripsi	Penjual dapat memperbaharui data pada operator tertentu yang diinginkan sesuai dengan yang ditetapkan oleh server penyedia pulsa
Kondisi Awal	Data untuk setiap operator telah tersimpan pada aplikasi dan penjual telah berhasil login
Kondisi Akhir	Data untuk setiap operator telah diperbarui, disesuaikan dan disimpan
Skenario	1. Penjual memilih data operator yang ingin diperbarui dan disesuaikan 2. Aplikasi menampilkan data operator yang dipilih 3. Penjual mengganti kode prefix 4. Penjual mengganti harga untuk nominal pembelian pulsa 5. Aplikasi memvalidasi kebenaran data yang dimasukkan penjual 6. Aplikasi menyimpan data operator
Skenario Alternatif	5.a. [Kode Prefix tidak diisi] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 3.

	5.a. [harga untuk tiap nominal tidak lengkap] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 4.
--	--

Tabel 3.5 mendeskripsikan alur kerja *use case* memperbaharui data operator agar sesuai dengan yang ditentukan oleh server penyedia pulsa dan tidak terjadi kesalahan dalam perhitungan total pendapatan.

Tabel 3. 6 Skenario *use case* mengatur nomor sms center

Nama use case	Mengatur Nomor SMS center
Aktor	Penjual
Deskripsi	Penjual melakukan pengaturan nomor SMS center yang digunakan sebagai nomor tujuan untuk mengirim permintaan pengisian pulsa pelanggan dan pengecekan saldo penjual pulsa
Kondisi Awal	Penjual telah berhasil login kedalam aplikasi
Kondisi Akhir	Nomor SMS center tersimpan
Skenario	1. Penjual memasukkan nomor SMS center yang telah didapat dari server penyedia pulsa 2. Aplikasi menyimpan nomor SMS center
	3. Aplikasi menampilkan nomor SMS Center yang tersimpan 4. Jika nomor yang telah terdaftar ingin dihapus, penjual memilih nomornya 5. Aplikasi menampilkan pesan konfirmasi penghapusan 6. Penjual mengkonfirmasi penghapusan untuk nomor yang dipilih 6. Aplikasi menghapus nomor yang telah dipilih
Skenario Alternatif	1.a. [nomor SMS center sudah ada] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 1 1.a. [nomor SMS center tidak diisi] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 1 5.a. [Penjual tidak mengkonfirmasi penghapusan] kembali ke skenario 3

Tabel 3.6 mendeskripsikan alur kerja *use case* mengatur nomor SMS center sebagai nomor tujuan SMS ketika melakukan permintaan pengisian pulsa pelanggan dan pengecekan saldo penjual pulsa.

Tabel 3. 7 Skenario *use case* menyimpan pin transaksi

Nama use case	Menyimpan PIN Transaksi
Aktor	Penjual
Deskripsi	Penjual menyimpan PIN transaksi pengisian pulsa
Kondisi Awal	Penjual telah berhasil login kedalam aplikasi
Kondisi Akhir	PIN Transaksi tersimpan pada aplikasi
Skenario	1. Penjual memasukkan PIN transaksi baru dan konfirmasi PIN transaksi yang baru 2. Aplikasi memvalidasi kebenaran PIN transaksi baru dan konfirmasi PIN transaksi baru yang telah dimasukkan penjual 3. Aplikasi menyimpan PIN transaksi baru yang sudah dimasukkan
Skenario Alternatif	3.a. [PIN transaksi baru dan konfirmasi PIN transaksi baru tidak sesuai] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 1

Tabel 3.7 mendeskripsikan alur kerja *use case* menyimpan PIN Transaksi pengisian pulsa yang digunakan pada saat transaksi pengisian pulsa dan juga pengecekan saldo pulsa yang dimiliki penjual.

Tabel 3. 8 Skenario *use case* mengatur karakter pemisah

Nama use case	Mengatur karakter pemisah
Aktor	Penjual
Deskripsi	Penjual melakukan pemilihan karakter pemisah untuk transaksi
Kondisi Awal	Penjual telah berhasil login kedalam aplikasi
Kondisi Akhir	Aplikasi menyimpan karakter pemisah
Skenario	1. Aplikasi menampilkan pilihan karakter pemisah 2. Penjual memilih karakter pemisah yang tersedia 3. Aplikasi menyimpan karakter pemisah yang sudah dipilih
Skenario Alternatif	-

Tabel 3.8 mendeskripsikan alur kerja *use case* mengatur karakter pemisah yang digunakan pada kode transaksi permintaan pengisian pulsa pelanggan.

Tabel 3. 9 Skenario *use case* mengganti pin transaksi

Nama use case	Mengganti PIN Transaksi
Aktor	Penjual
Deskripsi	Penjual mengganti PIN transaksi pengisian pulsa yang sebelumnya telah terdaftar pada aplikasi
Kondisi Awal	Penjual telah berhasil login kedalam aplikasi
Kondisi Akhir	PIN Transaksi telah terganti dan tersimpan pada aplikasi
Skenario	1. Penjual memasukkan PIN transaksi yang lama, PIN transaksi baru, dan konfirmasi PIN transaksi yang baru 2. Aplikasi memvalidasi kebenaran PIN transaksi lama, PIN transaksi baru, dan konfirmasi PIN transaksi baru yang telah dimasukkan penjual 3. Aplikasi menyimpan PIN transaksi baru yang sudah dimasukkan
Skenario Alternatif	3.a [PIN lama salah] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 1 3.a. [PIN transaksi baru dan konfirmasi PIN transaksi baru tidak sesuai] Aplikasi menampilkan pesan kesalahan. Kembali ke skenario 1

Tabel 3.9 mendeskripsikan alur kerja *use case* mengganti PIN transaksi pengisian pulsa.

Tabel 3. 10 Skenario *use case request* pengisian pulsa

Nama use case	<i>Request</i> pengisian pulsa
Aktor	Penjual
Deskripsi	Penjual melakukan transaksi pengisian pulsa pulsa pelanggan
Kondisi Awal	Penjual telah login kedalam aplikasi, pengaturan operator, SMS center, PIN transaksi, dan karakter pemisah telah tersimpan
Kondisi Akhir	Pelanggan menerima pulsa yang telah dibeli, transaksi pengiriman pulsa tersimpan
Skenario	1. Penjual memasukkan nomor pelanggan 2. Penjual memilih operator pulsa yang disediakan oleh aplikasi berdasarkan pengaturan operator yang telah tersimpan

	3. Penjual memilih nominal pulsa yang tersedia
	4. Penjual memilih nomor SMS center yang akan digunakan berdasarkan nomor yang telah tersimpan
	5. Penjual memilih status pembayaran lunas
	6. Aplikasi memvalidasi kebenaran masukan penjual
	7. Aplikasi menampilkan pesan konfirmasi transaksi pengisian pulsa
	8. Penjual mengkonfirmasi transaksi pengisian pulsa
	9. Aplikasi mengirim permintaan pengisian pulsa ke server penyedia pulsa
	10. Aplikasi menyimpan data transaksi
Skenario Alternatif	5.a. [status pembayaran belum lunas] Penjual memasukkan nama pelanggan. Lanjut ke skenario 7
	6.a. [Data yang dimasukkan tidak lengkap] Aplikasi menampilkan pesan kesalahan. kembali ke skenario 1.
	9.a. [Service kirim SMS tidak diizinkan] SMS request pengisian pulsa tidak dikirim. Lanjut ke skenario 10

Tabel 3.10 mendeskripsikan alur kerja *use case Request* pengisian pulsa yang dilakukan pada saat ada transaksi pengisian pulsa pelanggan yang baru.

Tabel 3. 11 Skenario *use case* melihat riwayat transaksi

Nama use case	Melihat Riwayat Transaksi
Aktor	Penjual
Deskripsi	Penjual melakukan pengecekan riwayat transaksi dan total pendapatan yang telah didapatkan dari hasil penjualan pulsa
Kondisi Awal	Penjual telah login kedalam aplikasi
Kondisi Akhir	Aplikasi menampilkan semua data transaksi yang telah dilakukan oleh penjual
Skenario	1. Penjual memasukkan periode tanggal yang data transaksinya ingin dilihat
	2. Aplikasi mengambil data transaksi
	3. Aplikasi menghitung total pendapatan selama periode yang diminta penjual
	4. Aplikasi menampilkan riwayat transaksi
Skenario Alternatif	2.a [data transaksi yang diminta tidak ada] aplikasi menampilkan notifikasi bahwa data yang diminta tidak ada

Tabel 3.9 mendeskripsikan alur kerja *use case* melihat riwayat transaksi ketika penjual melakukan pengecekan terhadap transaksi-transaksi yang telah dilakukan sekaligus total pendapatan sesuai periode yang diminta.

Tabel 3. 12 Skenario *use case* menghapus riwayat transaksi

Nama use case	Menghapus riwayat transaksi
Aktor	Penjual
Deskripsi	Penjual menghapus riwayat transaksi
Kondisi Awal	Penjual telah login kedalam aplikasi
Kondisi Akhir	Data transaksi telah terhapus sesuai dengan yang dipilih oleh penjual
Skenario	1. Penjual memasukkan periode tanggal yang data transaksinya ingin dilihat
	2. Aplikasi mengambil data-data transaksi
	3. Aplikasi menghitung total pendapatan selama periode yang diminta penjual

	4. Aplikasi menampilkan riwayat transaksi
	5. Penjual memilih data riwayat transaksi yang ingin dihapus
	6. Aplikasi menampilkan pesan konfirmasi penghapusan
	7. Aplikasi menghapus data riwayat transaksi terpilih
Skenario Alternatif	6.a [batal menghapus] kembali ke skenario 1

Tabel 3.10 mendeskripsikan alur kerja *use case* menghapus riwayat transaksi.

Tabel 3. 13 Skenario *use case* melihat laporan hutang

Nama use case	Melihat Laporan Hutang
Aktor	Penjual
Deskripsi	Penjual melakukan pengecekan laporan hutang transaksi penjualan pulsa
Kondisi Awal	Penjual telah login kedalam aplikasi
Kondisi Akhir	Aplikasi menampilkan laporan pelanggan yang belum melunasi pembayaran
Skenario	1. Aplikasi mengambil data transaksi yang status pembayarannya belum lunas 2. Aplikasi menampilkan laporan hutang
Skenario Alternatif	-

Tabel 3.13 mendeskripsikan alur kerja *use case* melihat laporan hutang ketika penjual ingin melihat siapa saja pelanggan yang belum melunasi pembayaran.

Tabel 3. 14 Skenario *use case* mengganti status pembayaran

Nama use case	Mengganti Status Pembayaran
Aktor	Penjual
Deskripsi	Penjual mengganti status pembayaran dari pelanggan yang telah melunasi hutang
Kondisi Awal	Penjual telah login kedalam aplikasi, laporan hutang ditampilkan
Kondisi Akhir	Status pembayaran telah terganti dan disimpan
Skenario	1. Aplikasi menampilkan laporan hutang, yaitu transaksi yang statusnya belum dibayar 2. Penjual memilih data hutang yang akan diubah statusnya 3. Penjual mengubah status pembayaran pelanggan yang hutang menjadi lunas 4. Aplikasi menyimpan status yang sudah diganti dan data dipindahkan ke riwayat transaksi
Skenario Alternatif	-

Tabel 3.14 mendeskripsikan alur kerja *use case* mengganti status pembayaran yang dilakukan ketika pelanggan telah melunasi pembayaran pulsanya.

Tabel 3. 15 Skenario *use case* lihat pesan masuk

Nama use case	Lihat Pesan Masuk
Aktor	Penjual
Deskripsi	Penjual melihat pesan SMS yang masuk ke ponsel yang digunakan
Kondisi Awal	Penjual telah login kedalam aplikasi
Kondisi Akhir	Aplikasi menampilkan pesan SMS yang masuk ke ponsel penjual

Skenario	1. Aplikasi menampilkan pesan SMS masuk yang berhubungan dengan transaksi pengisian pulsa
Skenario Alternatif	1.a. [Service baca SMS tidak diizinkan] data SMS masuk tidak ditampilkan

Tabel 3.15 mendeskripsikan alur kerja *use case* lihat pesan masuk yang digunakan untuk melihat pesan SMS masuk yang berhubungan dengan transaksi pengisian pulsa ke ponsel penjual.

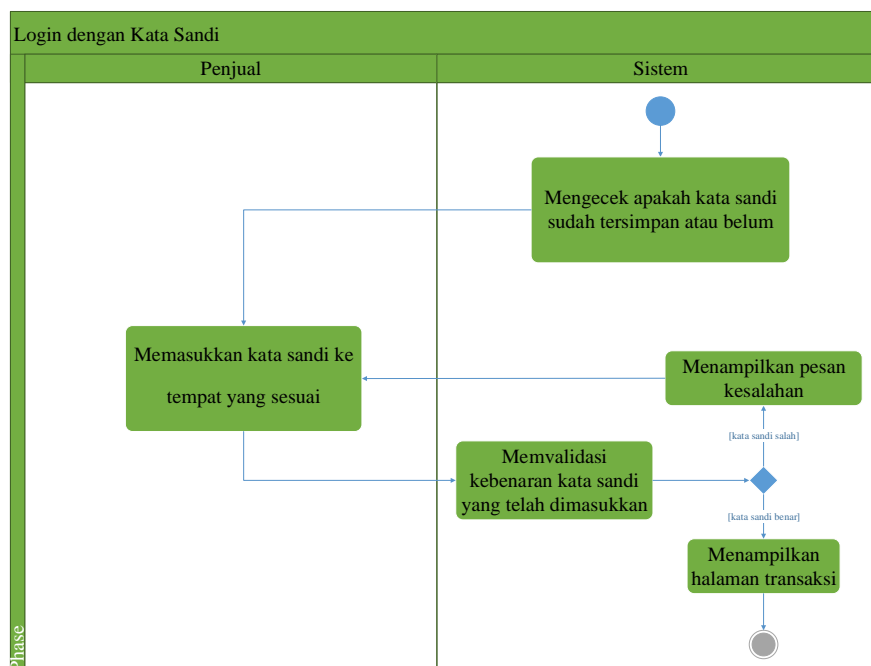
Tabel 3. 16 Skenario *use case* cek saldo

Nama use case	Cek Saldo
Aktor	Penjual
Deskripsi	Penjual melakukan pengecekan terhadap saldo pulsa yang dimiliki
Kondisi Awal	Penjual telah login kedalam aplikasi, PIN transaksi dan nomor SMS Center pengisian pulsa telah tersimpan pada aplikasi
Kondisi Akhir	SMS pemberitahuan sisa saldo penjual pulsa masuk ke ponsel penjual
Skenario	1. Penjual memilih nomor SMS Center 2. Aplikasi mengirimkan SMS permintaan cek saldo ke server penyedia pulsa 3. Aplikasi menerima SMS pemberitahuan sisa saldo penjual pulsa dari server penyedia pulsa
Skenario Alternatif	2.a. [Service kirim SMS tidak diizinkan] SMS <i>request</i> cek saldo tidak dikirim.

Tabel 3.16 mendeskripsikan alur kerja *use case* cek saldo penjual pulsa.

3.6 Activity Diagram

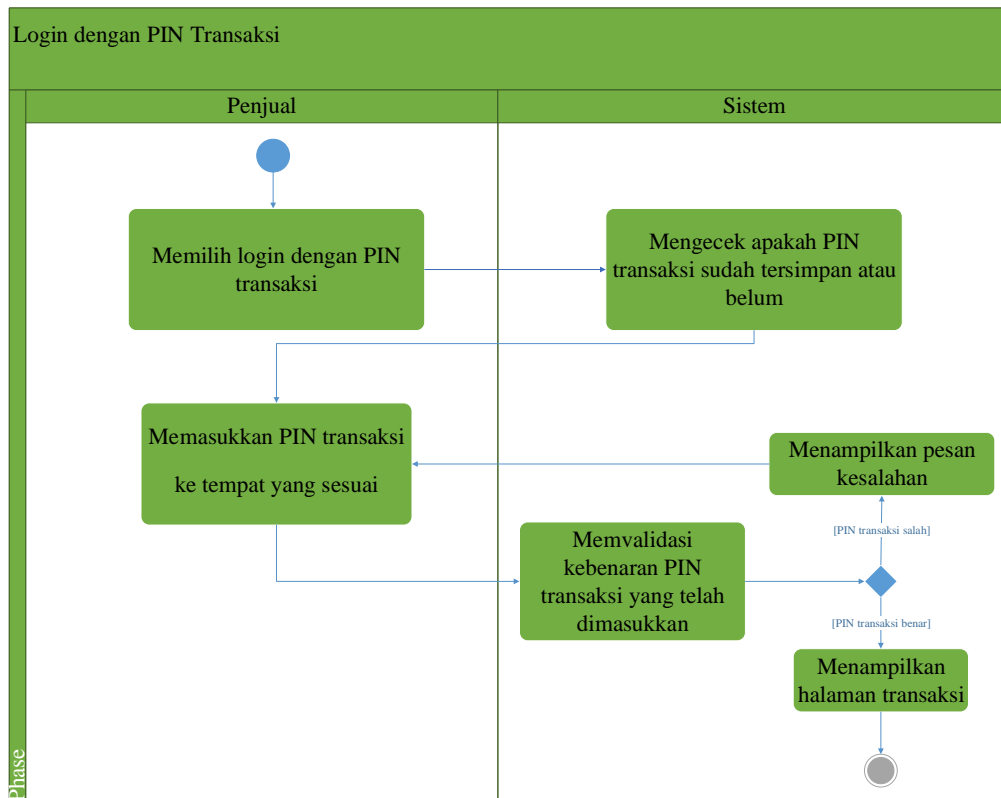
3.6.1 Activity diagram login dengan kata sandi



Gambar 3. 3 Activity diagram login dengan kata sandi

Berdasarkan Gambar 3.3 dapat menggambarkan aliran kerja pada saat penjual melakukan login pada Aplikasi Penjualan Pulsa Elektronik dengan menggunakan kata sandi dan terdapat beberapa kemungkinan proses tersebut akan gagal karena disebabkan beberapa kondisi tertentu.

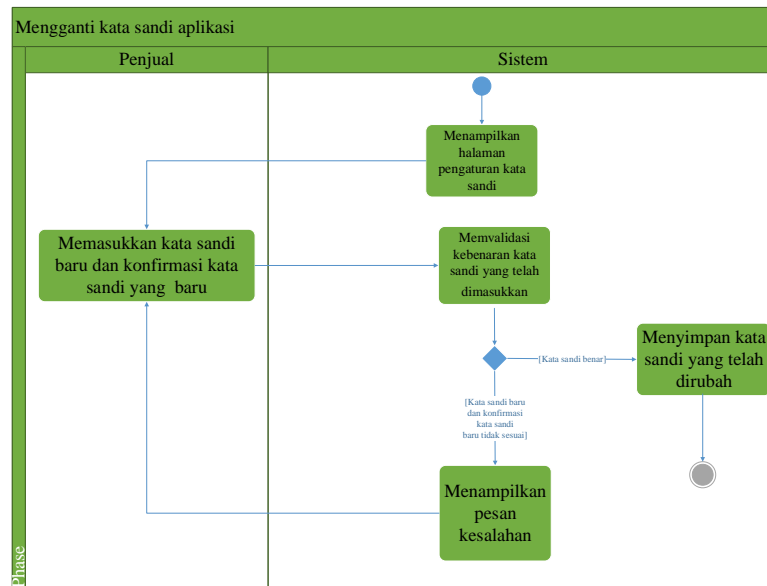
3.6.2 Activity diagram login dengan PIN transaksi



Gambar 3. 4 Activity diagram login dengan PIN transaksi

Berdasarkan Gambar 3.4 dapat menggambarkan aliran kerja pada saat penjual melakukan login pada Aplikasi Penjualan Pulsa Elektronik dengan menggunakan PIN transaksi pengisian pulsa dan terdapat beberapa kemungkinan proses tersebut akan gagal karena disebabkan beberapa kondisi tertentu.

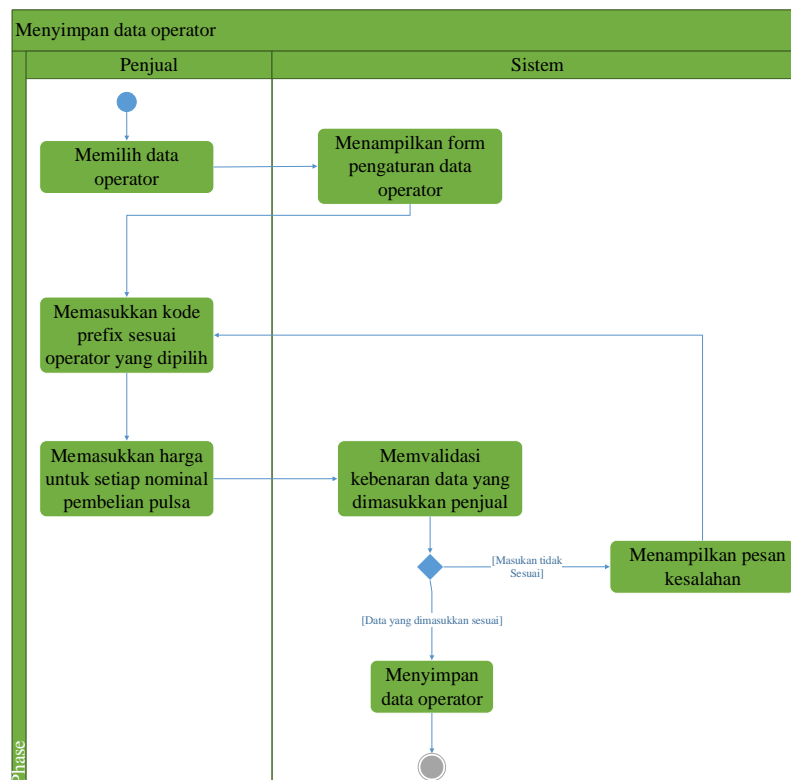
3.6.3 Activity diagram mengganti kata sandi aplikasi



Gambar 3. 5 Activity diagram mengganti kata sandi aplikasi

Berdasarkan Gambar 3.5 menggambarkan aliran kerja pada saat penjual mengganti kata sandi aplikasi.

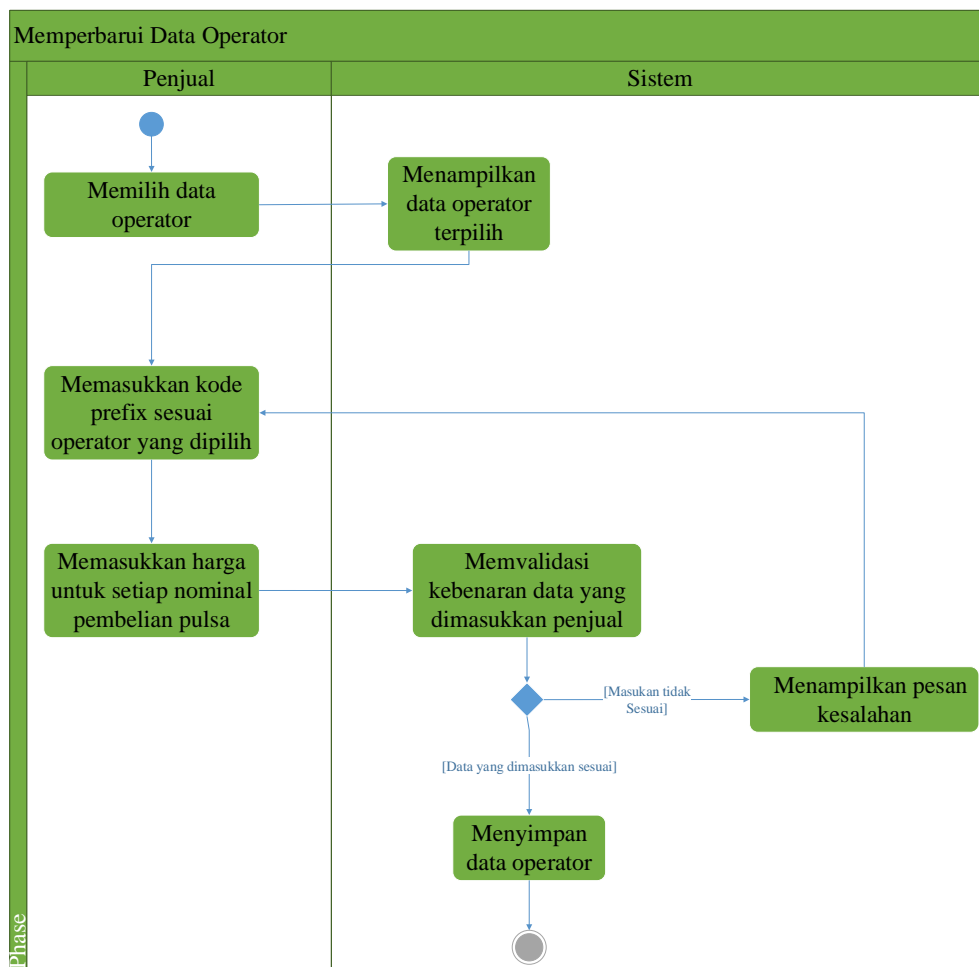
3.6.4 Activity diagram menyimpan data operator



Gambar 3. 6 Activity diagram memperbaharui data operator

Berdasarkan Gambar 3.6 menggambarkan aliran kerja pada saat penjual melakukan pengaturan data operator yang berisi kode prefix, dan harga untuk setiap nominal pembelian pulsa. Pengaturan ini berguna untuk format transaksi pengisian pulsa yang sesuai dengan yang ditetapkan oleh server penyedia pulsa dan juga untuk kebutuhan perhitungan pendapatan penjual.

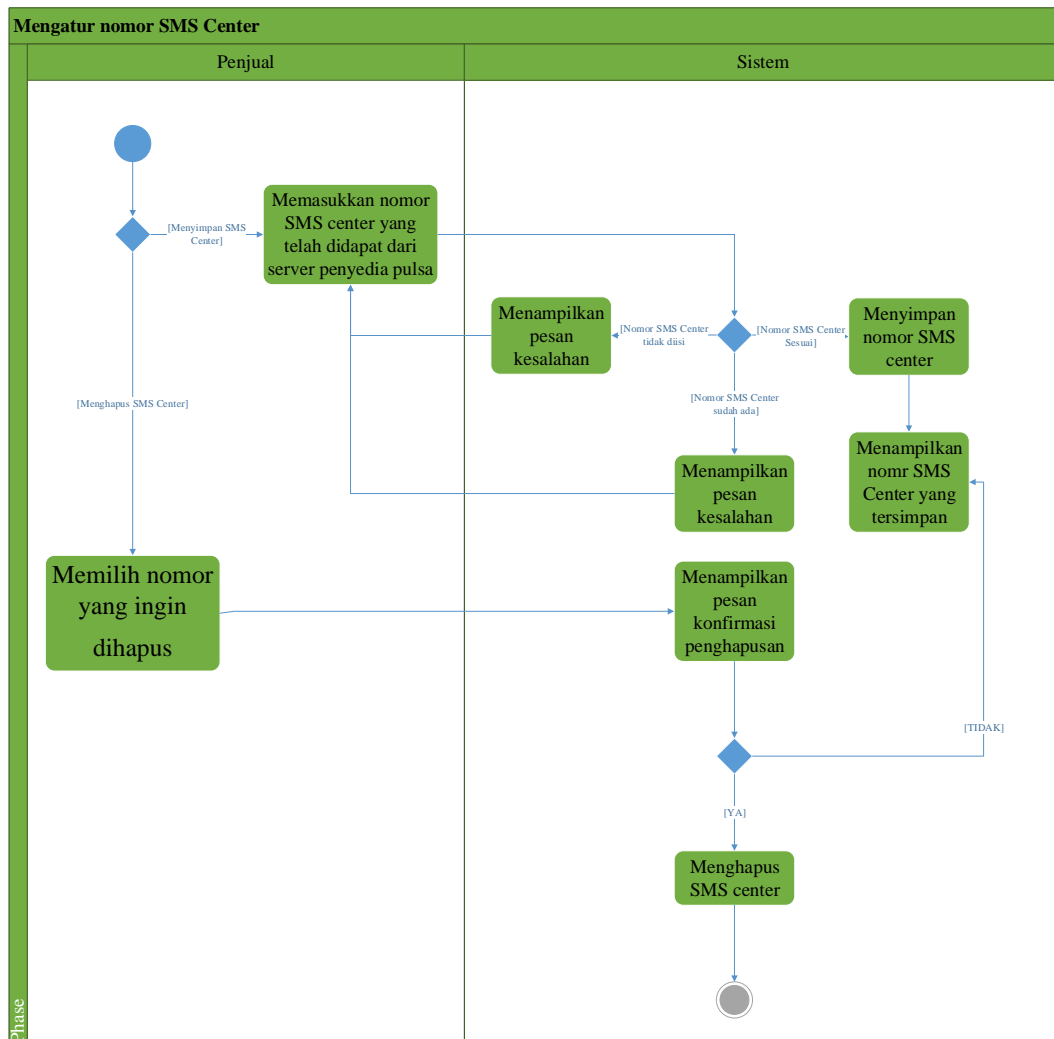
3.6.5 Activity diagram memperbaharui data operator



Gambar 3.7 Activity diagram memperbaharui data operator

Berdasarkan Gambar 3.7 menggambarkan aliran kerja pada saat penjual mengganti data operator yang berisi kode prefix, dan harga untuk setiap nominal pembelian pulsa. Pengaturan ini berguna untuk format transaksi pengisian pulsa yang sesuai dengan yang ditetapkan oleh server penyedia pulsa dan juga untuk kebutuhan perhitungan pendapatan penjual.

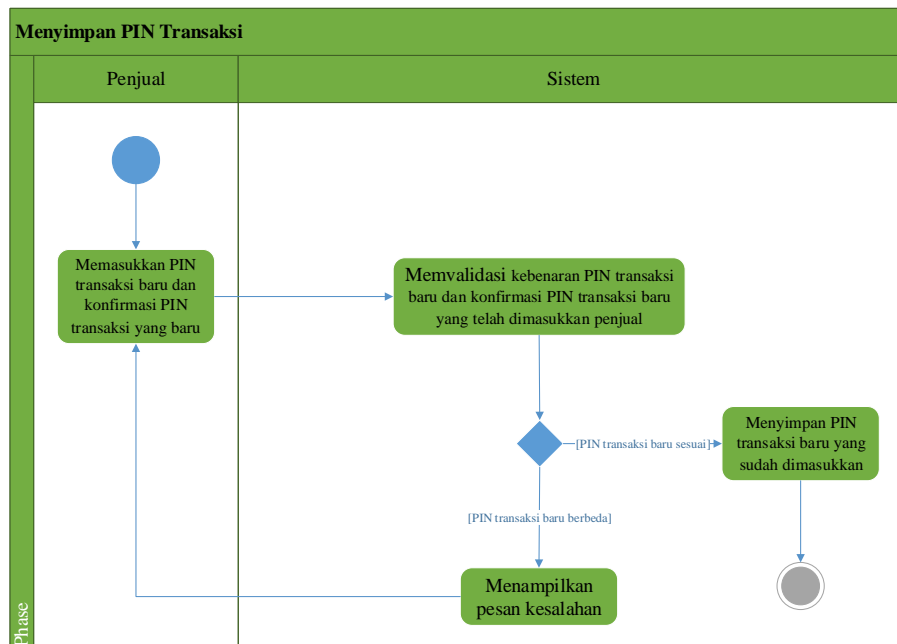
3.6.6 Activity diagram mengatur nomor SMS center



Gambar 3. 8 Activity diagram mengatur nomor SMS center

Berdasarkan Gambar 3.8, menggambarkan aliran kerja pada saat penjual melakukan pengaturan nomor SMS center yang akan digunakan sebagai nomor tujuan pada saat akan mengirimkan permintaan pengisian pulsa pembeli. Apabila nomor SMS center sudah terdaftar pada aplikasi, maka aplikasi akan menampilkan pesan kesalahan.

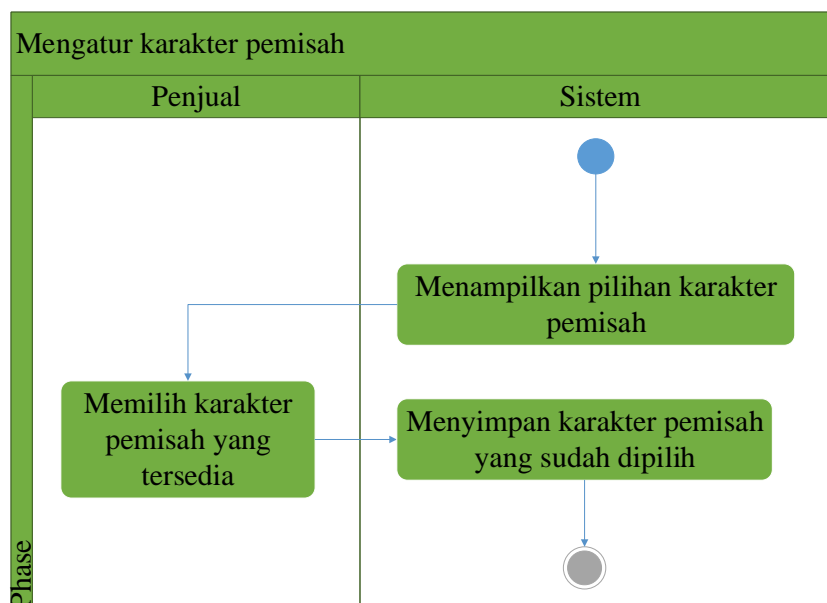
3.6.7 Activity diagram menyimpan PIN transaksi



Gambar 3. 9 Activity diagram menyimpan PIN transaksi

Berdasarkan Gambar 3.9 menggambarkan proses aliran kerja pada saat penjual menyimpan atau mendaftarkan PIN transaksi pengisian pulsa ke dalam aplikasi.

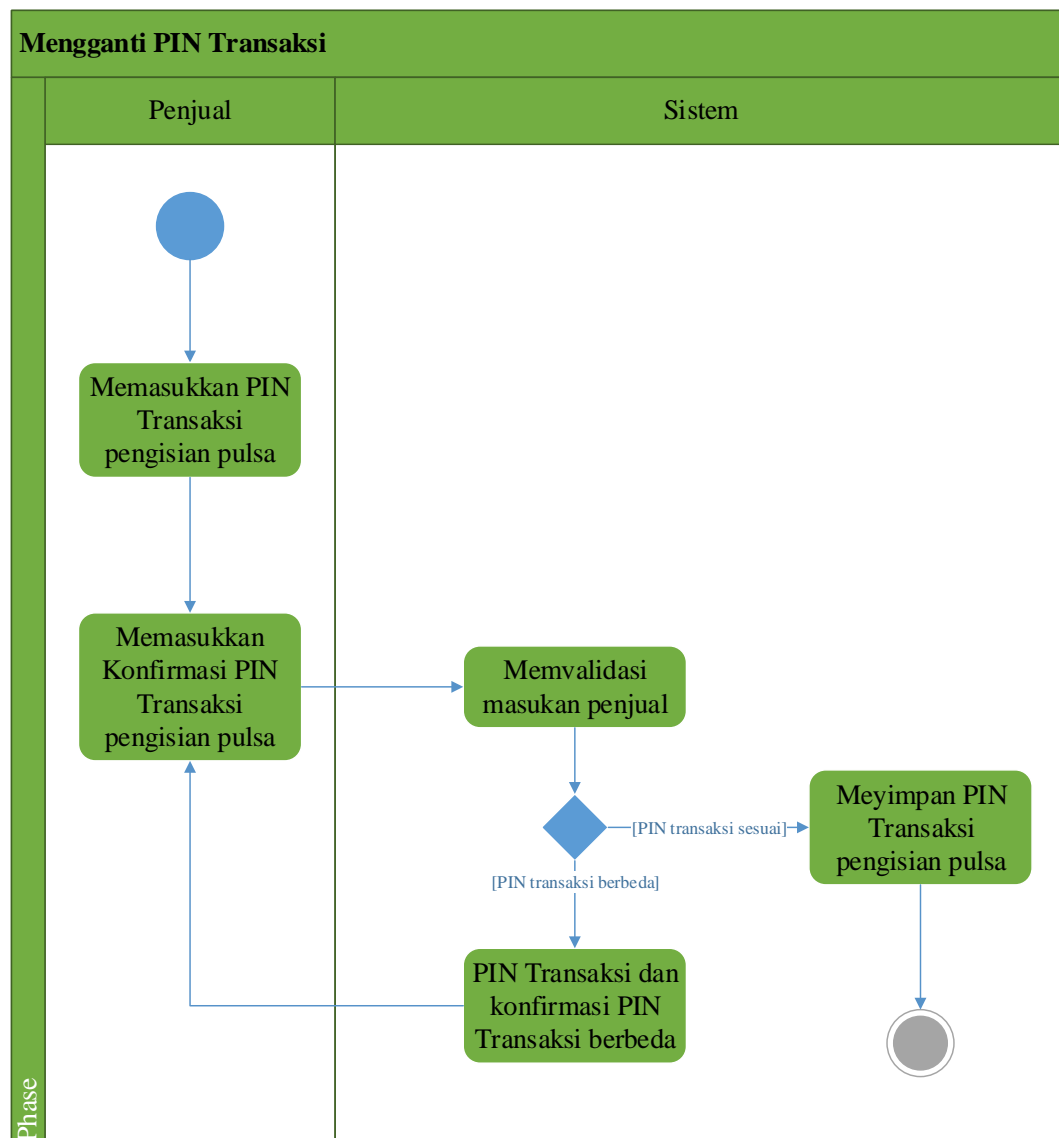
3.6.8 Activity diagram mengatur karakter pemisah



Gambar 3. 10 Activity diagram mengatur karakter pemisah

Berdasarkan Gambar 3.10, menunjukkan proses aliran kerja pada saat penjual melakukan pengaturan pada karakter pemisah yang digunakan pada format kode transaksi pengisian pulsa.

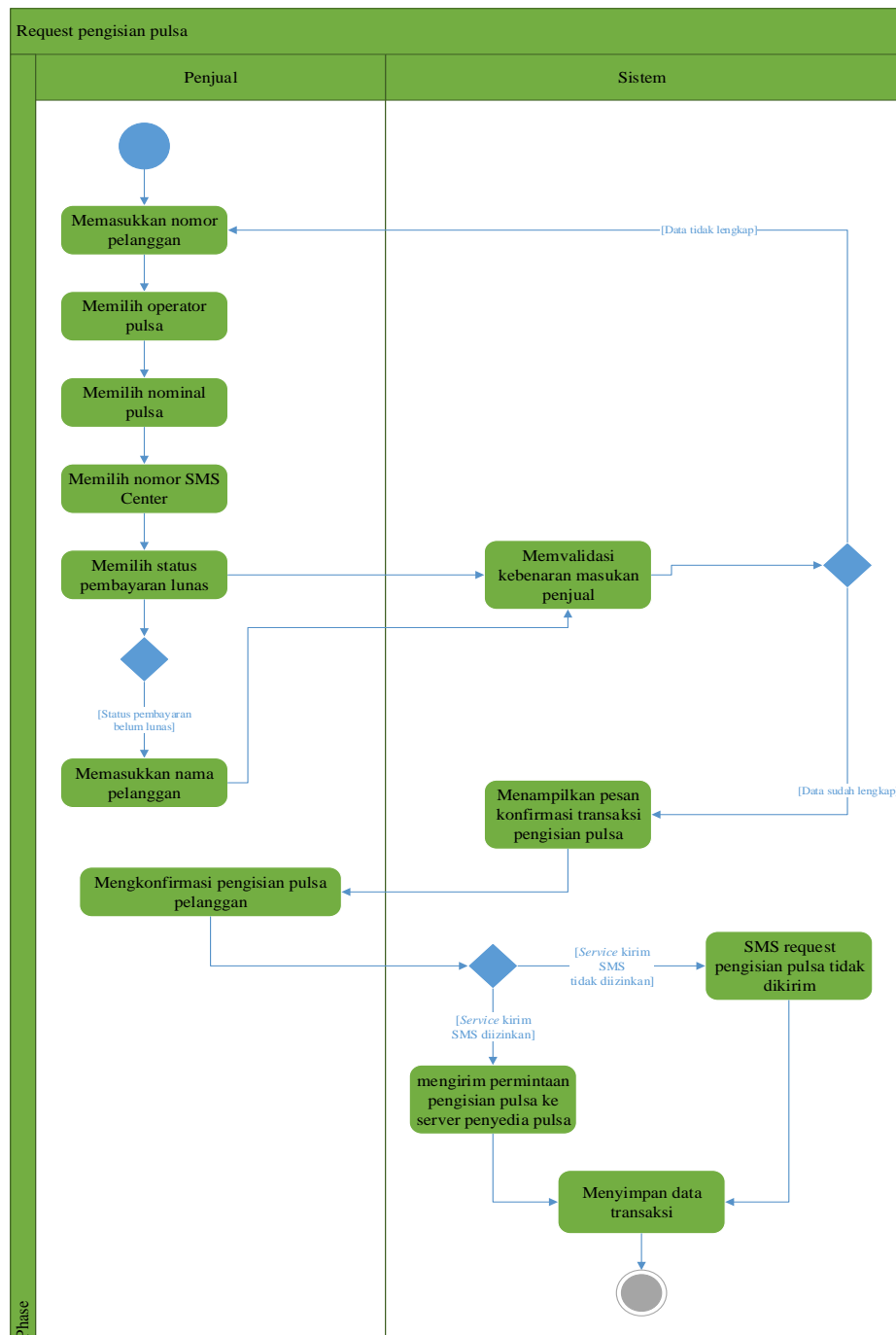
3.6.9 Activity diagram mengganti pin transaksi



Gambar 3. 11 Activity diagram mengganti PIN transaksi

Berdasarkan Gambar 3.11, menunjukkan proses aliran kerja pada saat penjual melakukan pengubahan PIN Transaksi pengisian pulsa pelanggan.

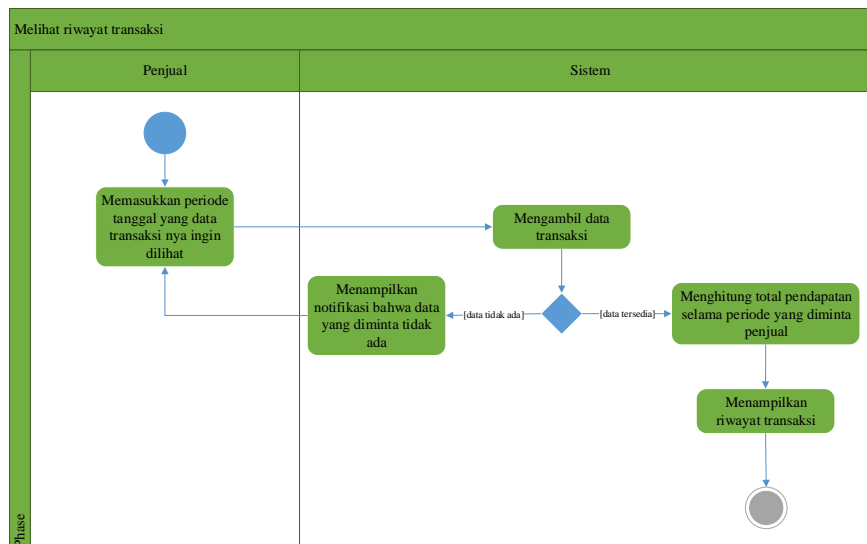
3.6.10 Activity diagram request pengisian pulsa



Gambar 3. 12 Activity diagram request pengisian pulsa

Berdasarkan Gambar 3.12, dapat dilihat aliran kerja pada saat melakukan proses permintaan pengisian pulsa oleh penjual ke nomor tujuan (pelanggan). Dan terdapat beberapa kemungkinan proses tersebut akan gagal karena disebabkan beberapa kondisi tertentu.

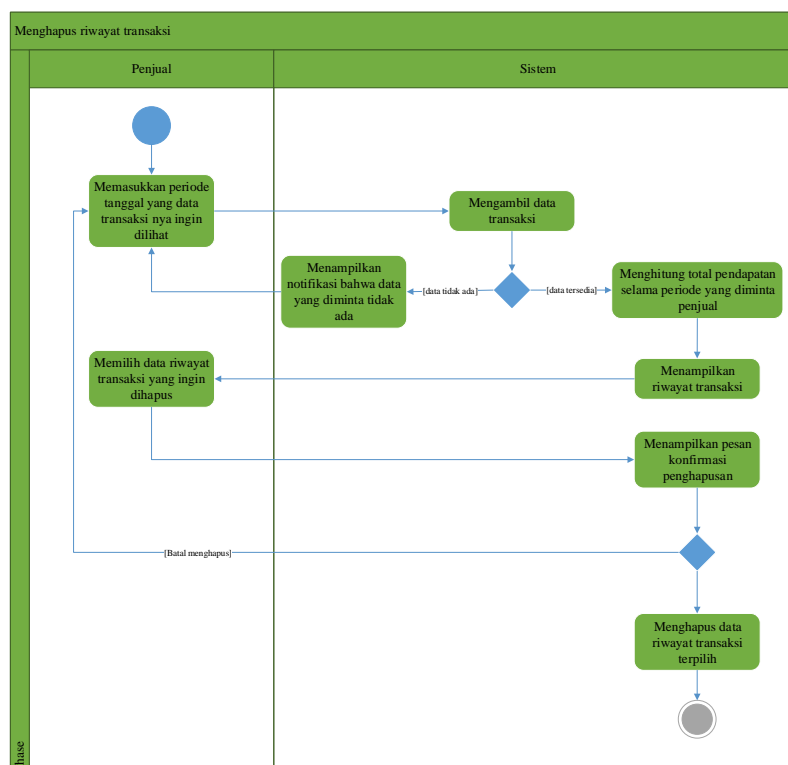
3.6.11 Activity diagram melihat riwayat transaksi



Gambar 3. 13 Activity diagram melihat riwayat transaksi

Berdasarkan Gambar 3.13, menggambarkan proses aliran kerja pada saat penjual melihat riwayat transaksi pengiriman pulsa dan total pendapatan yang diterima dalam periode waktu tertentu.

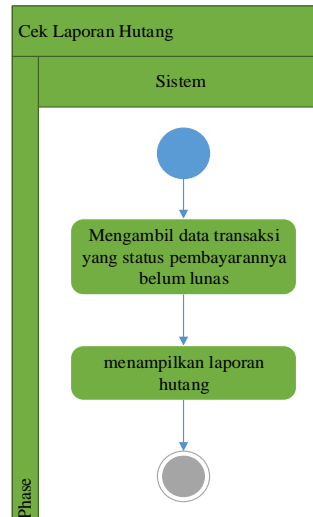
3.6.12 Activity diagram menghapus riwayat transaksi



Gambar 3. 14 Activity diagram menghapus riwayat transaksi

Berdasarkan Gambar 3.14, menunjukkan proses aliran kerja pada saat penjual menghapus riwayat transaksi pengiriman pulsa.

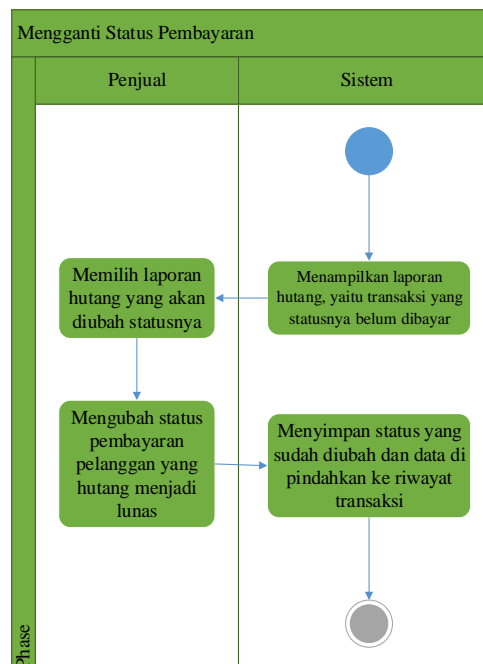
3.6.13 Activity diagram cek laporan hutang



Gambar 3. 15 Activity diagram cek laporan hutang

Berdasarkan Gambar 3.15, menunjukkan proses aliran kerja pada saat penjual melakukan pengecekan laporan hutang untuk melihat data pembeli yang belum melunasi pembayaran pembelian pulsa.

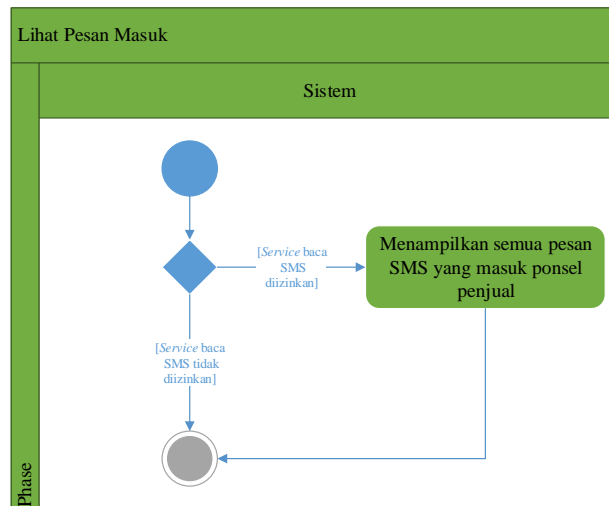
3.6.14 Activity diagram mengganti status pembayaran



Gambar 3. 16 Activity diagram mengganti status pembayaran

Berdasarkan Gambar 3.16, menunjukkan proses aliran kerja pada saat penjual melakukan perubahan atau memperbaharui status pembayaran pembeli yang sudah melunasi hutangnya.

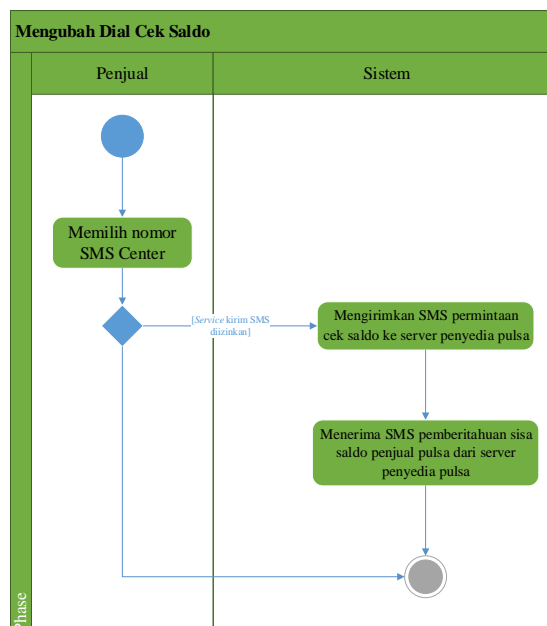
3.6.15 Activity diagram lihat pesan masuk



Gambar 3. 17 Activity diagram use case lihat pesan masuk

Berdasarkan Gambar 3.17 menunjukkan proses aliran kerja pada saat penjual melakukan pengecekan pesan-pesan SMS yang telah diterima dari server penyedia pulsa.

3.6.16 Activity diagram cek saldo

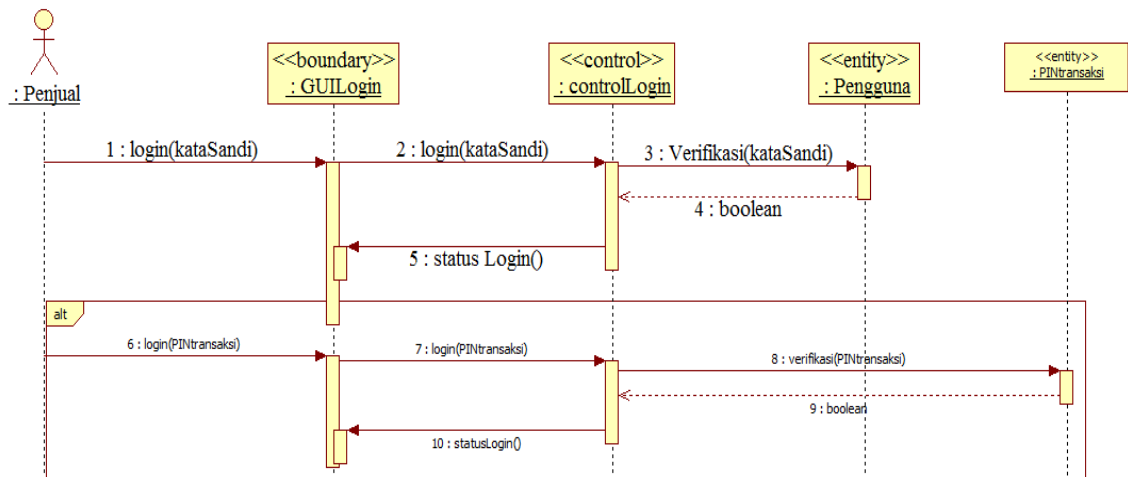


Gambar 3. 18 Activity diagram use case cek saldo

Berdasarkan Gambar 3.18, menunjukkan proses aliran kerja pada saat penjual melakukan pengecekan terhadap sisa saldo pulsa yang masih dimiliki.

3.7 Sequence Diagram

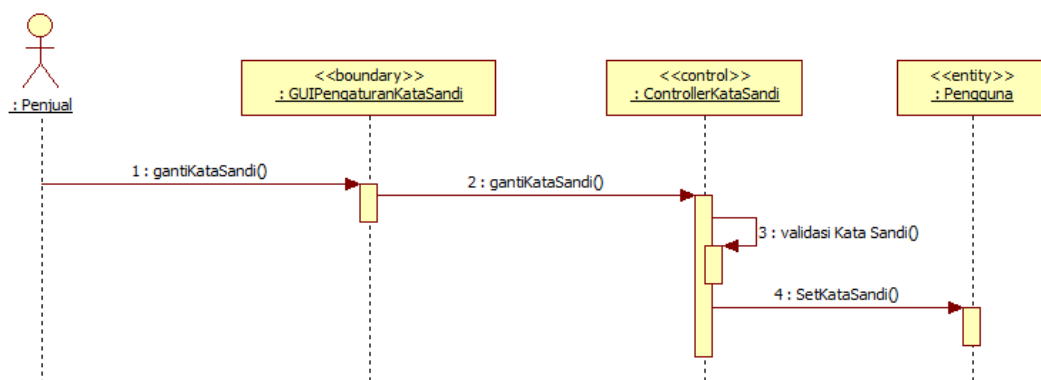
3.7.1 Sequence diagram login



Gambar 3. 19 Sequence diagram login

Gambar 3.19 memodelkan proses login penjual pulsa dimana untuk login, penjual memiliki opsi apakah ingin login dengan menggunakan kata sandi aplikasi atau dengan menggunakan PIN transaksi pengisian pulsa.

3.7.2 Sequence diagram mengganti kata sandi

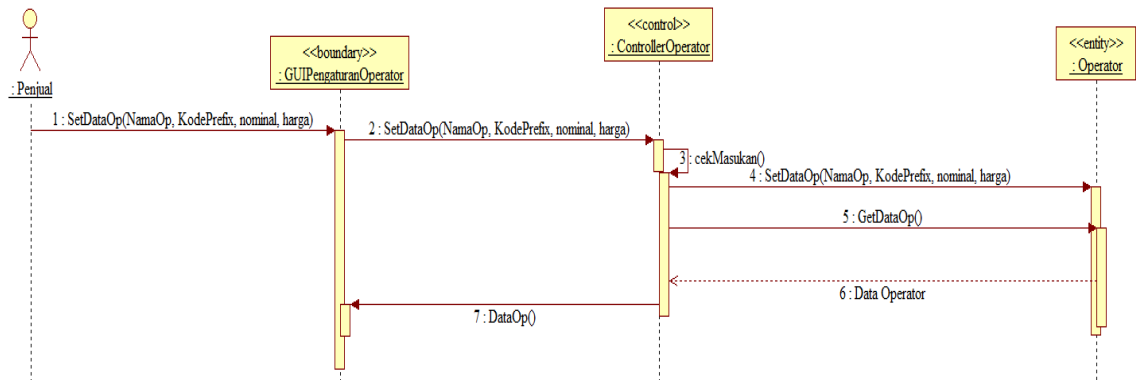


Gambar 3. 20 Sequence diagram mengganti kata sandi

Gambar 3.20 memodelkan proses yang terjadi pada saat penjual mengganti kata sandi aplikasi. Untuk mengubahnya, penjual harus memasukkan kata sandi aplikasi yang lama, kata sandi aplikasi yang baru, dan konfirmasi kata sandi yang baru. Jika

semuanya sudah benar, maka aplikasi akan memproses perubahan kata sandi aplikasi.

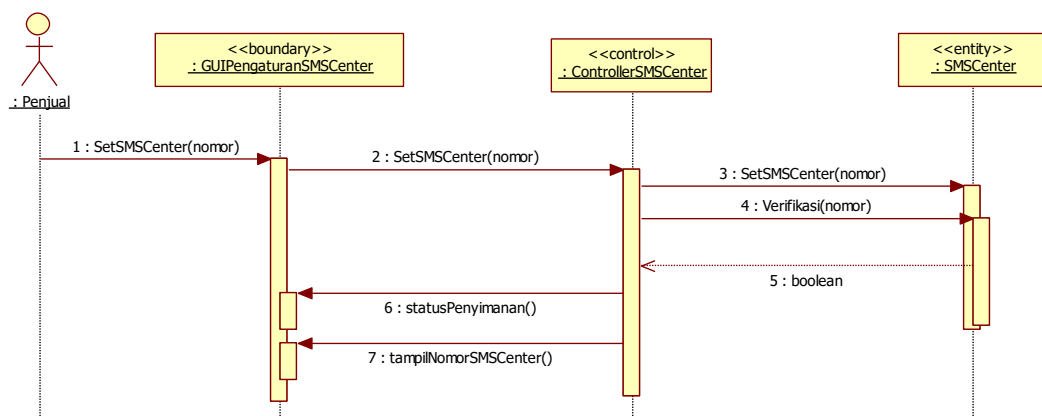
3.7.3 Sequence diagram memperbaharui data operator



Gambar 3. 21 Sequence diagram memperbaharui data operator

Gambar 3.21 memodelkan proses yang terjadi pada saat penjual melakukan pembaruan data operator untuk kebutuhan kode format transaksi pengisian pulsa yang sesuai dan perhitungan total pendapatan.

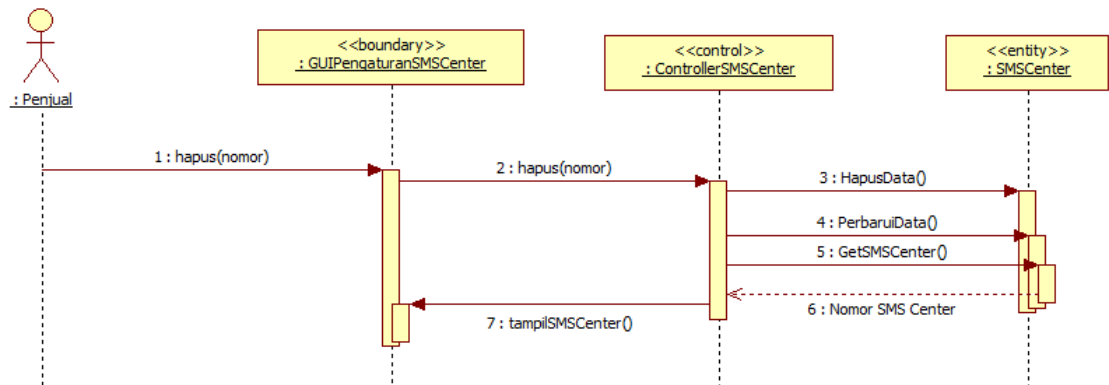
3.7.4 Sequence diagram mengatur nomor SMS center



Gambar 3. 22 Sequence diagram mengatur nomor SMS center

Gambar 3.22 memodelkan proses yang terjadi pada saat penjual mengatur nomor SMS center sebagai nomor tujuan permintaan pengisian pulsa dan permintaan cek sisa saldo penjual pulsa.

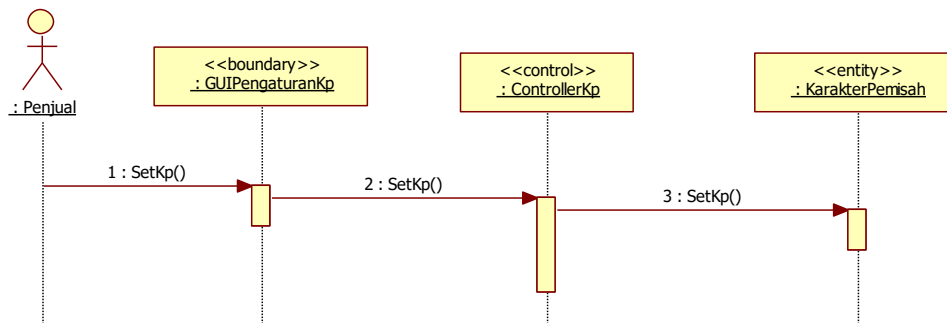
3.7.5 Sequence diagram menghapus sms center



Gambar 3. 23 Sequence diagram menghapus SMS center

Gambar 3.23 memodelkan proses yang terjadi pada saat penjual menghapus nomor SMS center yang sudah tidak dibutuhkan lagi.

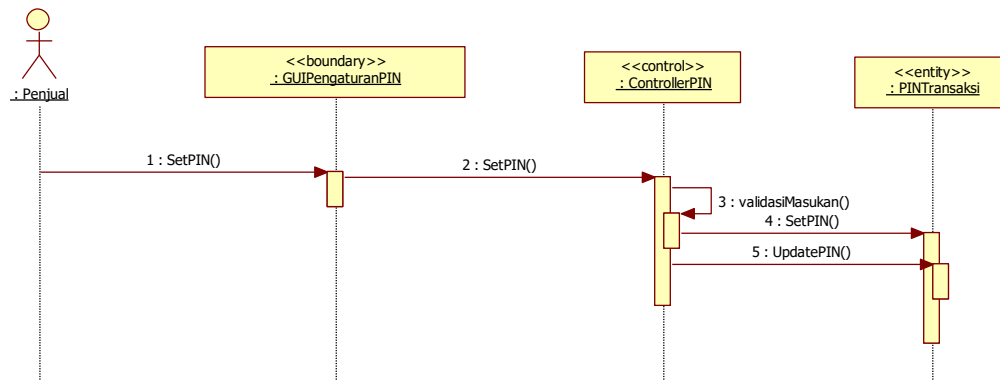
3.7.6 Sequence diagram menyimpan karakter pemisah



Gambar 3. 24 Sequence diagram mengatur karakter pemisah

Gambar 3.24 memodelkan proses yang terjadi pada saat penjual melakukan pengaturan karakter pemisah yang digunakan pada format kode transaksi pengisian pulsa.

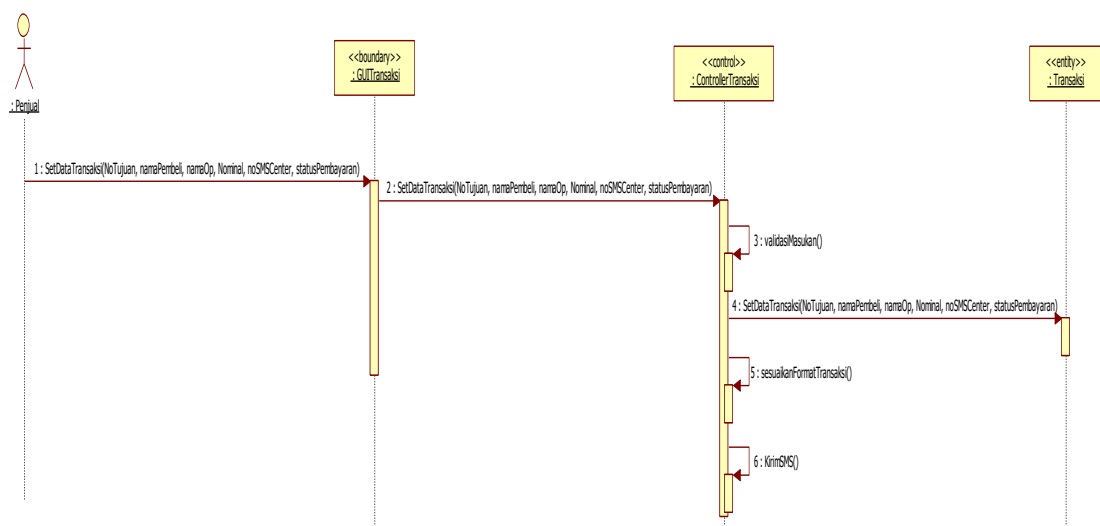
3.7.7 Sequence diagram mengganti PIN transaksi



Gambar 3. 25 Sequence diagram mengganti PIN transaksi

Gambar 3.25 memodelkan proses yang terjadi pada saat penjual mengganti PIN transaksi pengisian pulsa. Untuk mengubahnya, penjual harus memasukkan kata PIN transaksi yang lama, PIN transaksi yang baru, dan konfirmasi PIN transaksi yang baru. Jika semuanya sudah benar, maka aplikasi akan memproses perubahan PIN transaksi pengisian pulsa.

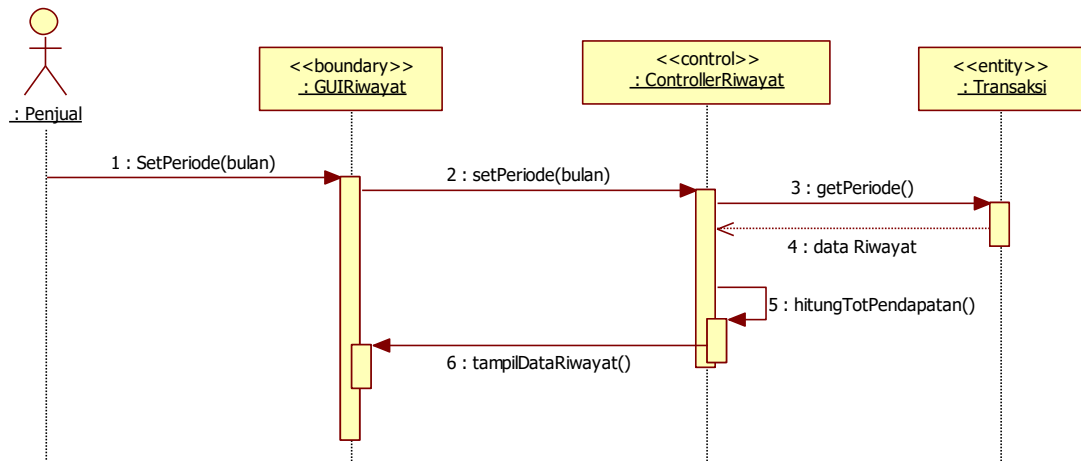
3.7.8 Sequence diagram request pengisian pulsa



Gambar 3. 26 Sequence diagram request pengisian pulsa

Gambar 3.26 memodelkan proses yang terjadi pada saat penjual melakukan transaksi pengisian pulsa.

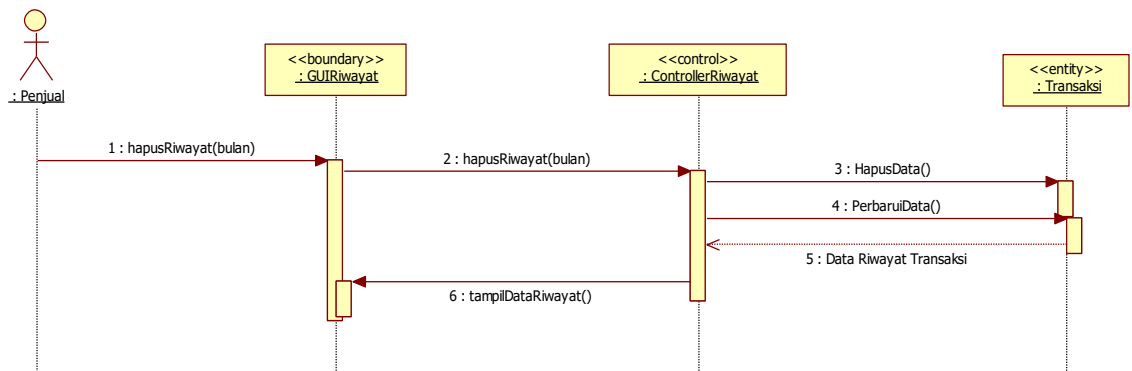
3.7.9 Sequence diagram melihat riwayat transaksi



Gambar 3. 27 Sequence diagram melihat riwayat transaksi

Gambar 3.27 memodelkan proses yang terjadi pada saat penjual ingin melihat riwayat transaksi pengisian pulsa pelanggan sekaligus mengecek total pendapatan yang telah diperolehnya.

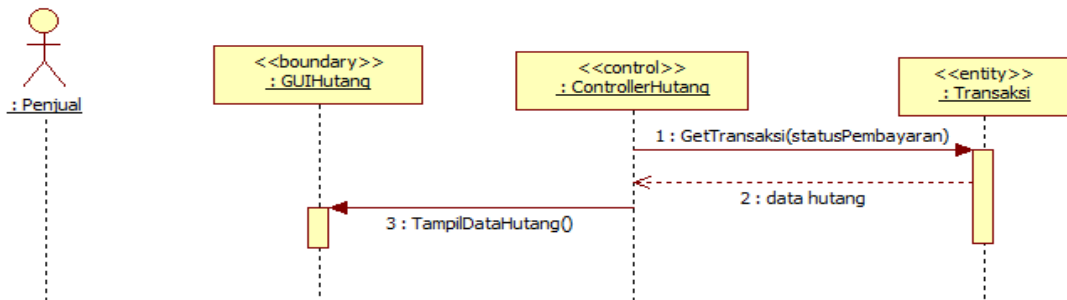
3.7.10 Sequence diagram menghapus riwayat transaksi



Gambar 3. 28 Sequence diagram menghapus riwayat transaksi

Gambar 3.28 memodelkan proses yang terjadi pada saat penjual ingin menghapus riwayat transaksi pengisian pulsa pelanggan.

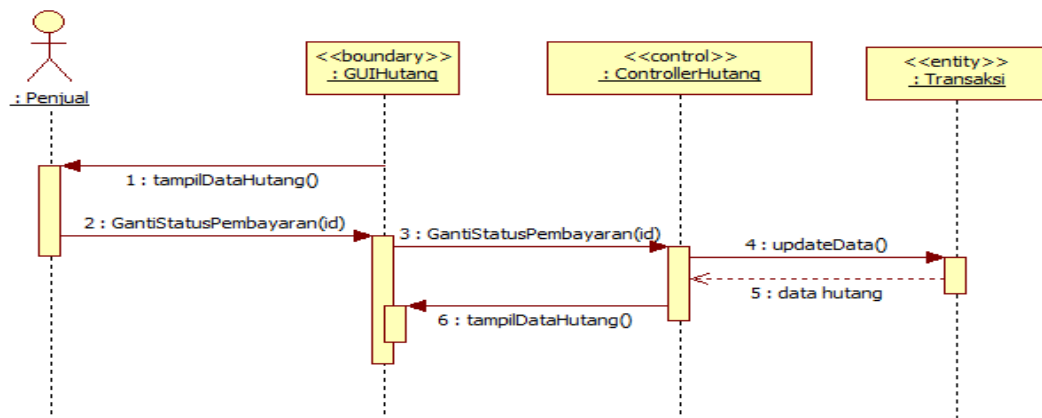
3.7.11 Sequence diagram melihat laporan hutang



Gambar 3. 29 Sequence diagram melihat laporan hutang

Gambar 3.29 memodelkan proses yang terjadi pada saat penjual ingin melihat laporan hutang untuk mengetahui siapa saja pelanggan yang belum melunasi pembayaran pulsanya.

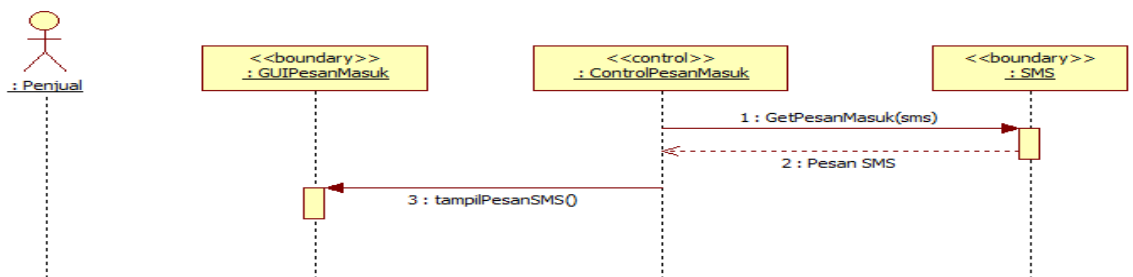
3.7.12 Sequence diagram mengganti status pembayaran



Gambar 3. 30 Sequence diagram mengganti status pembayaran

Gambar 3.30 memodelkan proses yang terjadi pada saat penjual mengganti status pembayaran pelanggan yang telah melunasi pembayaran uang pulsanya.

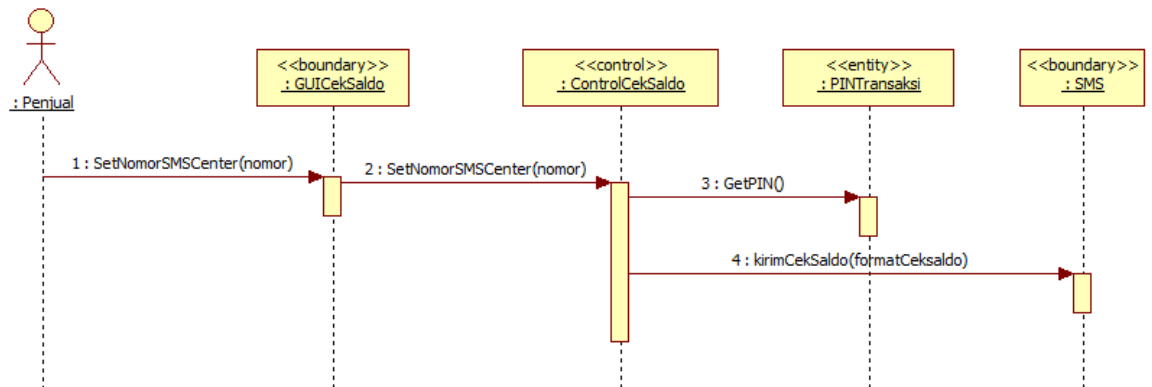
3.7.13 Sequence Diagram lihat pesan masuk



Gambar 3. 31 Sequence diagram lihat pesan masuk

Gambar 3.31 memodelkan proses yang terjadi pada saat penjual melihat pesan masuk yang diterima dari server penyedia pulsa.

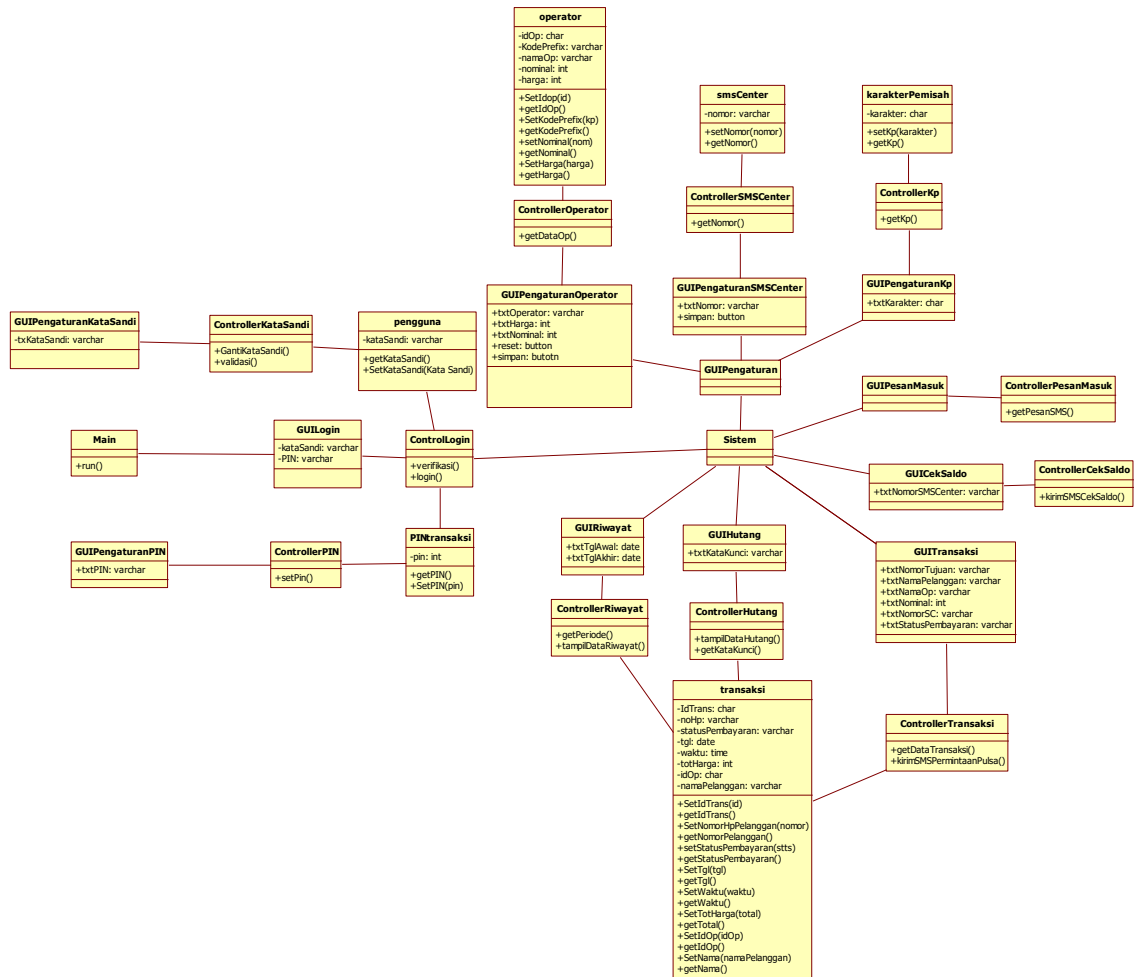
3.7.16 Sequence Diagram cek saldo



Gambar 3. 32 Sequence diagram cek saldo

Gambar 3.32 memodelkan proses yang terjadi pada saat penjual melakukan pengecekan sisa saldonya.

3.8 Class Diagram

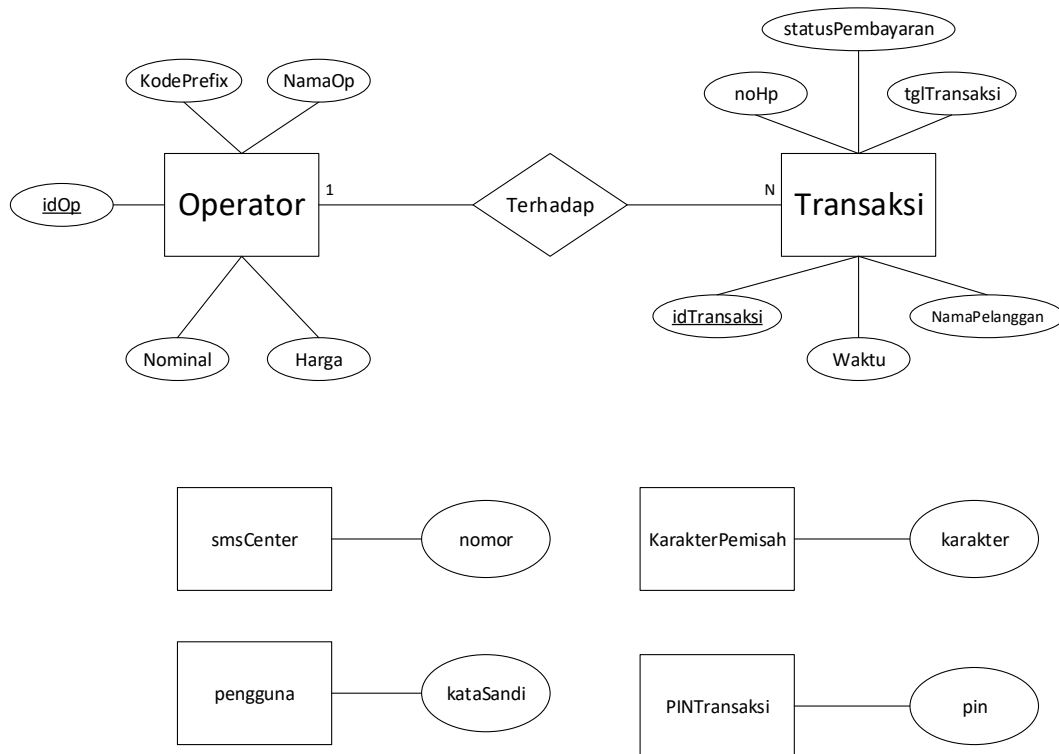


Gambar 3. 33 Class diagram aplikasi

Gambar 3.33 menggambarkan kelas-kelas yang ada pada aplikasi penjualan pulsa elektronik berbasis Android yang terdiri dari beberapa kelas GUI (*Graphical User Interface*), beberapa kelas *controller*, dan beberapa kelas entitas serta memperlihatkan hubungan diantara kelas-kelas tersebut lengkap dengan operasi-operasi dan atribut-atributnya.

3.9 Perancangan Basis Data

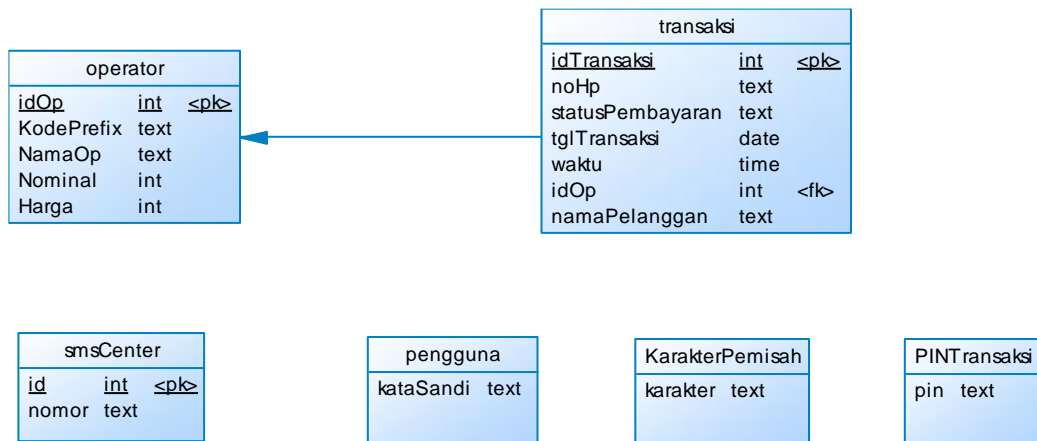
3.9.1 Entity Relationship Diagram (ERD)



Gambar 3. 34 *Entity relationship diagram*

Gambar 3.34 menunjukkan entitas-entitas beserta atribut serta relasi antar entitas yang ada pada aplikasi penjualan pulsa elektronik berbasis Android.

3.9.2 Physical Data Modelling (PDM)

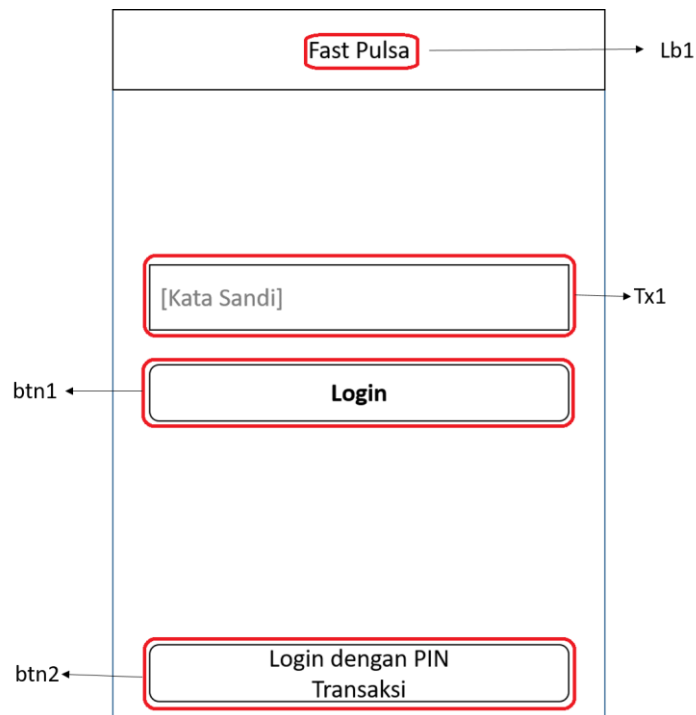


Gambar 3. 35 Physical data modelling

Gambar 3.35 menunjukkan tabel beserta atribut yang ada pada aplikasi penjualan pulsa elektronik berbasis Android. Terdapat 3 tabel yang disimpan kedalam basis data dan 3 tabel yang disimpan dalam bentuk file. Tabel yang disimpan kedalam basis data adalah tabel operator yang digunakan untuk menyimpan data kode prefix, nama operator, nominal, dan harga pulsa, tabel transaksi yang digunakan untuk menyimpan data transaksi dimana id pulsa pada tabel transaksi digunakan untuk mengambil data pulsa yang ada pada tabel pulsa, dan tabel SMS Center yang digunakan untuk menyimpan nomor SMS Center sebagai nomor tujuan permintaan pengiriman pulsa pelanggan pada saat proses transaksi pengiriman pulsa dan juga untuk permintaan pengecekan sisa saldo penjual. Sedangkan tabel yang disimpan dalam bentuk file adalah tabel pengguna yang digunakan untuk menyimpan kata sandi aplikasi, tabel karakterPemisah yang digunakan untuk menyimpan karakter pemisah pada format kode transaksi pengisian pulsa, dan tabel PINtransaksi yang digunakan untuk menyimpan pin transaksi pengisian pulsa.

3.10 Perancangan Antarmuka

3.10.1 Perancangan antarmuka login



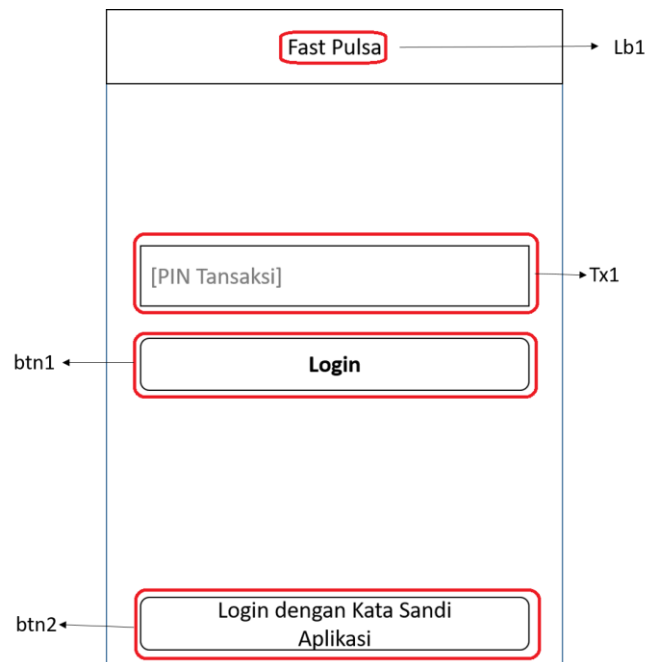
Gambar 3. 36 Perancangan antarmuka login dengan kata sandi aplikasi

Gambar 3.36 merupakan gambar perancangan antarmuka login. Penjual sebagai pengguna harus memasukkan kata sandi aplikasi agar bisa login kedalam aplikasi.

Tabel 3. 17 Deskripsi perancangan antarmuka login dengan kata sandi aplikasi

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast Pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Tx1	Edit text	Kata sandi	Tempat untuk memasukkan kata sandi aplikasi
btn1	Button	Login	Jika diklik maka aplikasi akan memvalidasi masukan penjual. Jika masukan benar, maka penjual akan mendapat hak akses aplikasi dan aplikasi menampilkan halaman transaksi pulsa
Btn2	Button	Login dengan PIN transaksi	Jika diklik maka aplikasi akan menampilkan halaman login dengan menggunakan PIN transaksi

Disajikan oleh Tabel 3.17 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka login menggunakan kata sandi aplikasi.



Gambar 3. 37 Perancangan antarmuka login dengan PIN transaksi

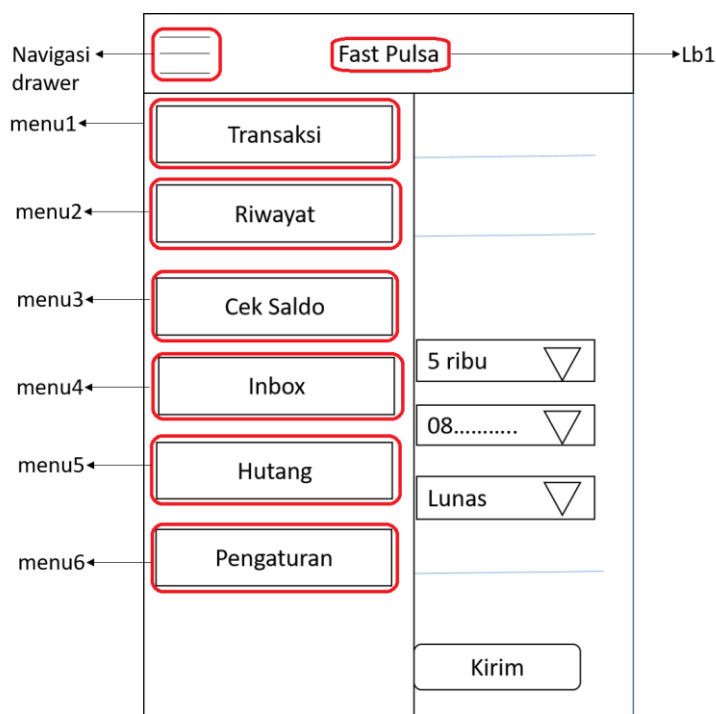
Gambar 3.37 merupakan gambar perancangan antarmuka login dengan PIN transaksi dimana penjual harus memasukkan PIN transaksi yang telah terdaftar pada aplikasi agar dapat login kedalam aplikasi.

Tabel 3. 18 Deskripsi perancangan antarmuka login dengan PIN transaksi

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast Pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Tx1	Edit text	PIN transaksi	Tempat untuk memasukkan PIN transaksi
btn1	Button	Login	Jika diklik maka aplikasi akan memvalidasi masukan penjual. Jika masukan benar, maka penjual akan mendapat hak akses aplikasi dan aplikasi menampilkan halaman transaksi pulsa
Btn2	Button	Login dengan Kata Sandi Aplikasi	Jika diklik maka aplikasi akan menampilkan halaman login dengan menggunakan kata sandi aplikasi

Disajikan oleh Tabel 3.18 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka login menggunakan PIN transaksi pengisian pulsa.

3.10.2 Perancangan antarmuka menu utama



Gambar 3. 38 Perancangan antarmuka menu utama

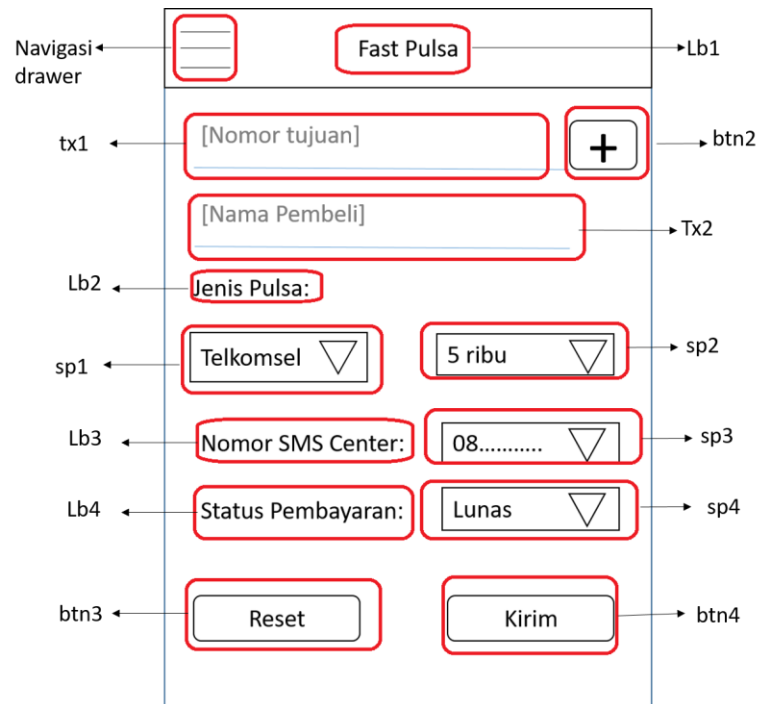
Gambar 3.38 merupakan gambar perancangan antarmuka menu utama dari aplikasi penjualan pulsa elektronik berbasis android.

Tabel 3. 19 Deskripsi perancangan antarmuka menu utama

Id_objek	Jenis	Nama	Keterangan
Navigasi drawer	Navigasi drawer	Button navigation drawer	Jika diklik maka akan menampilkan menu utama yang ditampilkan dengan navigation drawer
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Menu1	Menu	Transaksi	Jika diklik maka akan menampilkan halaman transaksi
Menu2	Menu	Riwayat	Jika diklik maka akan menampilkan halaman riwayat dari transaksi pengisian pulsa
Menu3	Menu	Cek Saldo	Jika diklik maka akan menampilkan halaman untuk mengirim SMS <i>request</i> cek saldo
Menu4	Menu	Inbox	Jika diklik maka akan menampilkan halaman pesan SMS masuk
Menu5	Menu	Hutang	Jika diklik maka akan menampilkan halaman hutang
Menu6	Menu	Pengaturan	Jika diklik maka akan menampilkan halaman pengaturan

Disajikan oleh Tabel 3.19 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka menu utama.

3.10.3 Perancangan antarmuka transaksi



Gambar 3. 39 Perancangan antarmuka transaksi

Gambar 3.39 merupakan gambar perancangan antarmuka halaman transaksi yang digunakan untuk memasukkan data transaksi pengisian pulsa pelanggan.

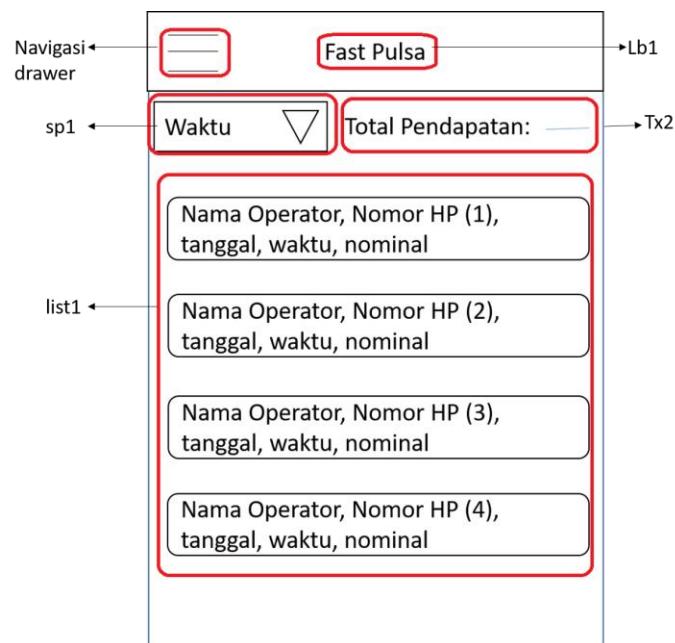
Tabel 3. 20 Deskripsi perancangan antarmuka transaksi

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Lb2	Label	Jenis Pulsa	Menunjukkan bahwa pilihan dibawahnya adalah data jenis pulsa
Lb3	Label	Nomor SMS Center	Menunjukkan bahwa data yang harus dipilih adalah data nomor SMS Center
Lb4	Label	Status Pembayaran	Menunjukkan bahwa data yang harus dipilih adalah data status pembayaran
Navigasi drawer	Navigasi drawer	Button navigation drawer	Jika diklik maka akan menampilkan menu utama yang ditampilkan dengan navigation drawer
btn2	Button	Import kontak	Jika diklik maka akan menampilkan aplikasi kontak bawaan ponsel android yang digunakan untuk memilih data kontak
btn3	Button	Reset	Jika diklik maka seluruh data yang sebelumnya telah diisi akan kembali ke kondisi awal sebelum data diisi
btn4	Button	Kirim	Jika diklik maka aplikasi akan memproses permintaan pengisian pulsa pembeli ke server penyedia pulsa

Id_objek	Jenis	Nama	Keterangan
tx1	edit text	Nomor tujuan	Merupakan tempat untuk menginputkan nomor pembeli pulsa
tx2	edit text	Nama pembeli	Merupakan tempat untuk menginputkan nama pembeli pulsa
Sp1	Spinner	Nama operator	Merupakan menu <i>dropdown</i> yang berisi nama operator untuk dipilih oleh penjual
Sp2	Spinner	Nominal pulsa	Merupakan menu <i>dropdown</i> yang berisi nominal pulsa yang tersedia dan harus dipilih oleh penjual
Sp3	Spinner	Nomor SMS Center	Merupakan menu <i>dropdown</i> yang berisi nomor SMS Center yang terdapat pada basis data aplikasi
Sp4	Spinner	Status pembayaran	Merupakan menu <i>dropdown</i> yang berisi status pembayaran telah lunas atau belum

Disajikan oleh Tabel 3.20 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka transaksi.

3.10.4 Perancangan antarmuka riwayat transaksi



Gambar 3. 40 Perancangan antarmuka riwayat transaksi

Gambar 3.40 merupakan gambar perancangan antarmuka riwayat transaksi yang digunakan untuk menunjukkan data riwayat transaksi beserta total pendapatan dalam periode waktu tertentu.

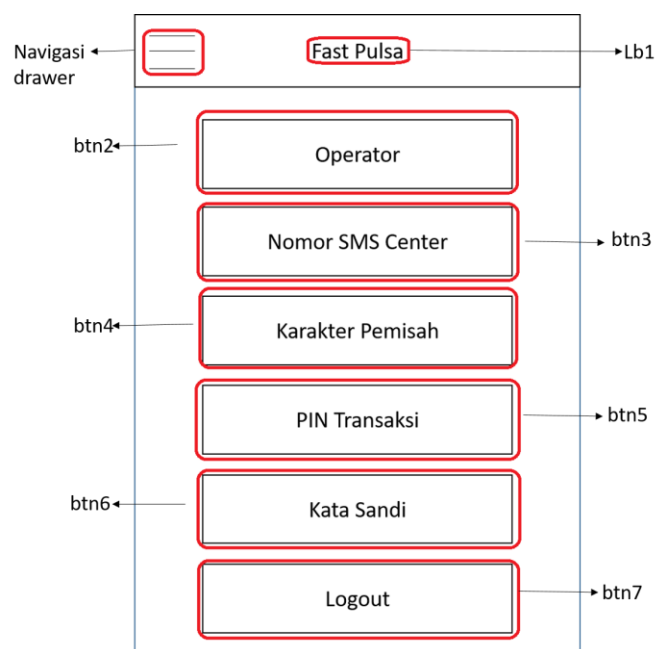
Tabel 3. 21 Deskripsi perancangan antarmuka riwayat transaksi

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa

Id_objek	Jenis	Nama	Keterangan
Navigasi drawer	Navigasi drawer	Button navigation drawer	Jika diklik maka akan menampilkan menu utama yang ditampilkan dengan navigation drawer
Sp1	Spinner	waktu	Untuk menginputkan pilihan rentang waktu riwayat transaksi
Tx2	edit text	Total pendapatan	Untuk menampilkan total pendapatan berdasarkan inputan waktu yang dimasukkan penjual
List1	list view	Data riwayat	Untuk menampilkan data riwayat transaksi penjualan pulsa yang tersimpan pada basis data

Disajikan oleh Tabel 3.21 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka riwayat transaksi.

3.10.5 Perancangan antarmuka menu pengaturan



Gambar 3. 41 Perancangan antarmuka menu pengaturan

Gambar 3.41 merupakan gambar perancangan antarmuka menu pengaturan. Penjual bisa memilih salah satu menu untuk melanjutkan ke pengaturan yang dipilih.

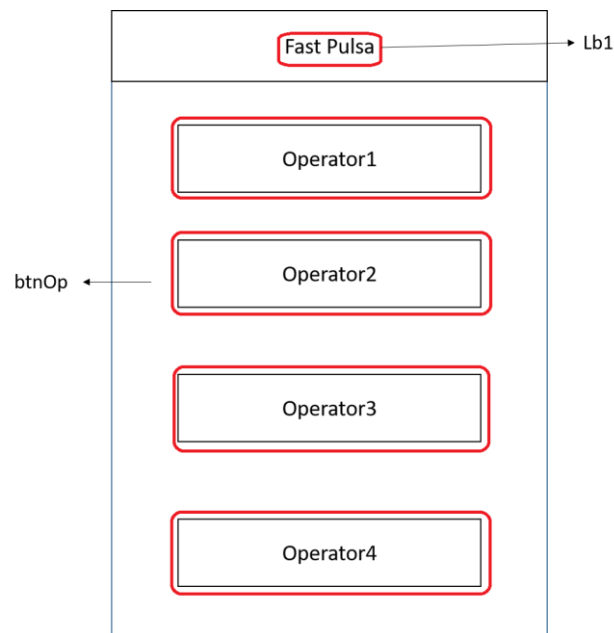
Tabel 3. 22 Deskripsi perancangan antarmuka menu pengaturan

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Navigasi drawer	Navigasi drawer	Button navigation drawer	Jika diklik maka akan menampilkan menu utama yang ditampilkan dengan navigation drawer
Btn2	Button	Operator	Jika diklik maka aplikasi akan menampilkan halaman pengaturan operator

Id_objek	Jenis	Nama	Keterangan
Btn3	Button	Nomor SMS Center	Jika diklik maka aplikasi akan menampilkan halaman pengaturan SMS Center
Btn4	Button	Karakter Pemisah	Jika diklik maka aplikasi akan menampilkan halaman pengaturan karakter pemisah
Btn5	Button	PIN Transaksi	Jika diklik maka aplikasi akan menampilkan halaman pengaturan PIN Transaksi
Btn6	Button	Kata sandi	Jika diklik maka aplikasi akan menampilkan halaman pengaturan kata sandi aplikasi
Btn7	Button	Logout	Jika diklik maka aplikasi akan menampilkan halaman login dan penjuall tidak mendapat hak akses aplikasi

Disajikan oleh Tabel 3.22 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka pengaturan.

3.10.6 Perancangan antarmuka menu utama pengaturan operator



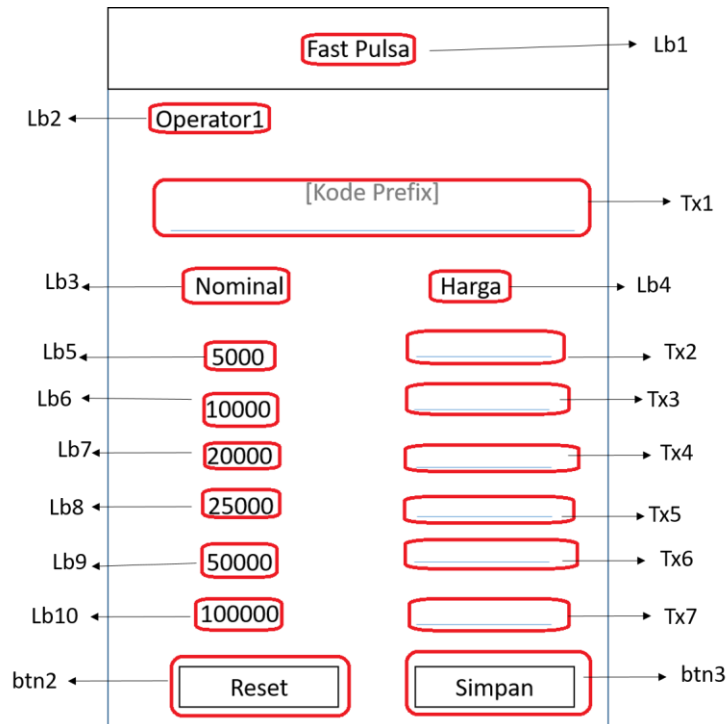
Gambar 3. 42 Perancangan antarmuka menu utama pengaturan operator

Gambar 3.42 merupakan gambar perancangan antarmuka menu utama pengaturan operator yang menunjukkan operator-operator pulsa yang tersedia pada aplikasi.

Tabel 3. 23 Deskripsi perancangan antarmuka menu utama pengaturan operator

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
btnOp	Button	Button nama operator	Jika diklik akan menampilkan halam pengaturan data untuk operator yang dipilih

Disajikan oleh Tabel 3.23 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka pengaturan operator.



Gambar 3. 43 Perancangan antarmuka pengaturan operator

Gambar 3.43 merupakan gambar perancangan antarmuka pengaturan operator yang digunakan untuk mengatur data tiap operator pulsa.

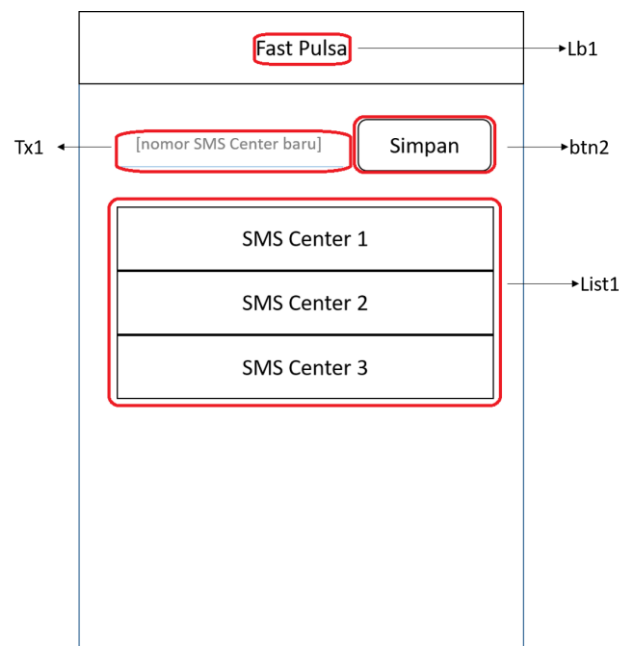
Tabel 3. 24 Deskripsi perancangan antarmuka pengaturan operator

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Lb2	Label	Nama operator	Menerangkan bahwa tampilan tersebut merupakan tampilan pengaturan untuk nama operator yang tertulis
Lb3	Label	Nominal	Menunjukkan bahwa yang ditampilkan dibawahnya adalah nominal pembelian pulsa
Lb4	Label	Harga	Menunjukkan bahwa yang seharusnya diinputkan dibawahnya adalah harga untuk nominal yang sesuai
Lb5	Label	5000	Menunjukkan bahwa yang seharusnya diinputkan adalah harga untuk nominal pembelian pulsa 5000
Lb6	Label	10000	Menunjukkan bahwa yang seharusnya diinputkan adalah harga untuk nominal pembelian pulsa 10000
Lb7	Label	20000	Menunjukkan bahwa yang seharusnya diinputkan adalah harga untuk nominal pembelian pulsa 20000
Lb8	Label	25000	Menunjukkan bahwa yang seharusnya diinputkan adalah harga untuk nominal pembelian pulsa 25000
Lb9	Label	50000	Menunjukkan bahwa yang seharusnya diinputkan adalah harga untuk nominal pembelian pulsa 50000

Id_objek	Jenis	Nama	Keterangan
Lb10	Label	100000	Menunjukkan bahwa yang seharusnya diinputkan adalah harga untuk nominal pembelian pulsa 10000
Tx1	edit text	Kode prefix	Tempat untuk memasukkan kode prefix dari operator terpilih
Tx2	edit text	Harga 5000	Tempat untuk menginputkan harga pulsa untuk nominal pembelian pulsa 5000
Tx3	edit text	Harga 10000	Tempat untuk menginputkan harga pulsa untuk nominal pembelian pulsa 10000
Tx4	edit text	Harga 20000	Tempat untuk menginputkan harga pulsa untuk nominal pembelian pulsa 20000
Tx5	edit text	Harga 25000	Tempat untuk menginputkan harga pulsa untuk nominal pembelian pulsa 25000
Tx6	edit text	Harga 50000	Tempat untuk menginputkan harga pulsa untuk nominal pembelian pulsa 50000
Tx7	edit text	Harga 100000	Tempat untuk menginputkan harga pulsa untuk nominal pembelian pulsa 10000
Tx8	edit text	Kode prefix	Tempat untuk menginputkan kode prefix dari nama operator yang sedang diatur
Btn9	Button	Button kembali	Jika diklik maka akan menampilkan halaman menu pemilihan operator yang ingin diatur
Btn2	Button	Reset	Jika diklik maka seluruh data yang sebelumnya telah diisi akan kembali ke kondisi awal sebelum data diisi
Btn3	Button	Simpan	Jika diklik maka yang telah diinputkan akan disimpan ke dalam basis data aplikasi

Disajikan oleh Tabel 3.24 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka pengaturan operator.

3.10.7 Perancangan antarmuka pengaturan SMS center



Gambar 3. 44 Perancangan antarmuka pengaturan SMS Center

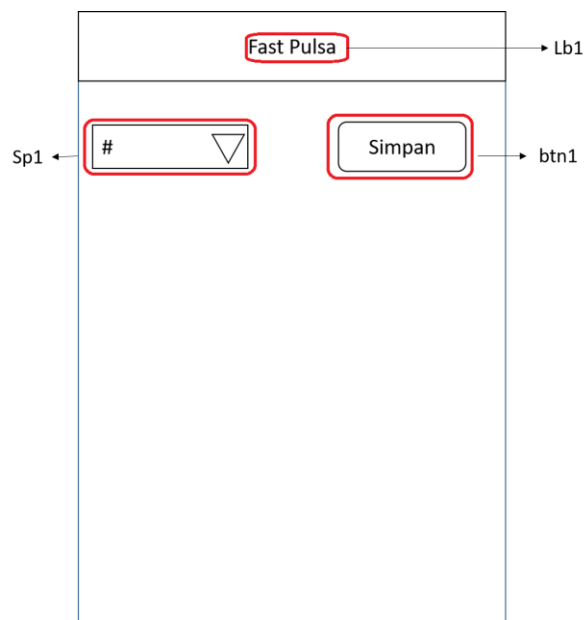
Gambar 3.44 merupakan gambar perancangan antarmuka pengaturan SMS Center yang digunakan sebagai nomor tujuan pengiriman permintaan pengisian pulsa pelanggan dan pengecekan saldo penjual pulsa. Penjual dapat memasukkan nomor SMS Center baru ataupun menghapus nomor SMS center yang telah tersimpan.

Tabel 3. 25 Deskripsi perancangan antarmuka pengaturan SMS Center

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Btn2	Button	Simpan	Jika diklik nomor SMS Center yang telah diinputkan akan ditambahkan ke basis data aplikasi
Tx1	edit text	Nomor SMS Center	Tempat untuk menginputkan nomor SMS Center baru
List1	list view	Data SMS Center	Menampilkan nomor-nomor SMS Center yang telah terdaftar pada basis data sitem

Disajikan oleh Tabel 3.25 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka pengaturan SMS Center.

3.10.8 Perancangan antarmuka pengaturan karakter pemisah



Gambar 3. 45 Perancangan antarmuka pengaturan karakter pemisah

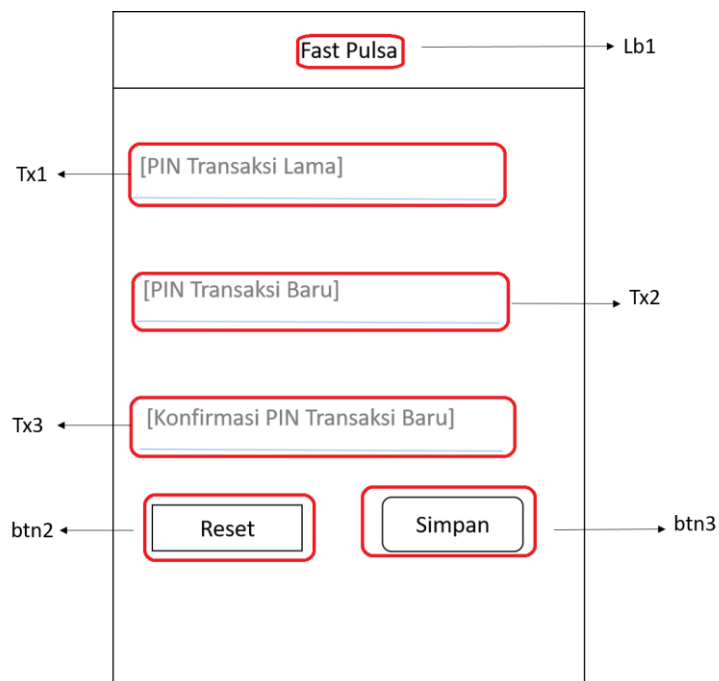
Gambar 3.45 merupakan gambar perancangan antarmuka pengaturan karakter pemisah yang digunakan dalam format kode transaksi pengisian pulsa pelanggan.

Tabel 3. 26 Deskripsi perancangan antarmuka pengaturan karakter pemisah

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Btn1	Button	Simpan	Jika diklik maka karakter pemisah yang telah dipilih akan tersimpan pada aplikasi
Sp1	Spinner	Karakter pemisah	Menampilkan karakter-karakter yang akan digunakan untuk memisah antar kata pada format permintaan pengisian pulsa pelanggan

Disajikan oleh Tabel 3.26 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka pengaturan karakter pemisah.

3.10.9 Perancangan antarmuka pengaturan PIN transaksi

**Gambar 3. 46 Perancangan antarmuka pengaturan PIN transaksi**

Gambar 3.46 merupakan gambar perancangan antarmuka pengaturan PIN transaksi yang digunakan untuk transaksi pengisian pulsa pelanggan, pengecekan saldo penjual pulsa, dan juga bisa digunakan untuk login kedalam aplikasi.

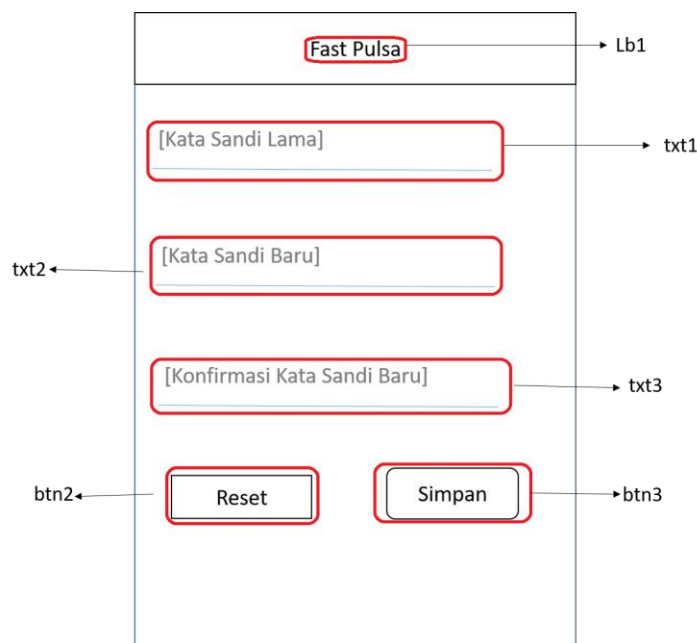
Tabel 3. 27 Deskripsi perancangan antarmuka pengaturan PIN transaksi

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Btn2	Button	Reset	Jika diklik maka data-data yang sebelumnya telah diisi akan terhapus dan edit text akan kembali kosong
Btn3	Button	Simpan	Jika diklik maka PIN Transaksi yang telah diatur akan tersimpan pada basis data aplikasi

Id_objek	Jenis	Nama	Keterangan
Tx1	edit text	PIN transaksi lama	Tempat untuk memasukkan PIN transaksi yang lama
Tx2	edit text	PIN transaksi baru	Tempat untuk memasukkan PIN transaksi yang baru
Tx3	editText	Konfirmasi PIN transaksi baru	Tempat untuk memasukkan konfirmasi PIN transaksi yang baru

Disajikan oleh Tabel 3.27 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka pengaturan PIN transaksi.

3.10.11 Perancangan antarmuka pengaturan kata sandi



Gambar 3. 47 Perancangan antarmuka pengaturan kata sandi

Gambar 3.45 merupakan gambar perancangan antarmuka pengaturan kata sandi aplikasi yang digunakan untuk login kedalam aplikasi.

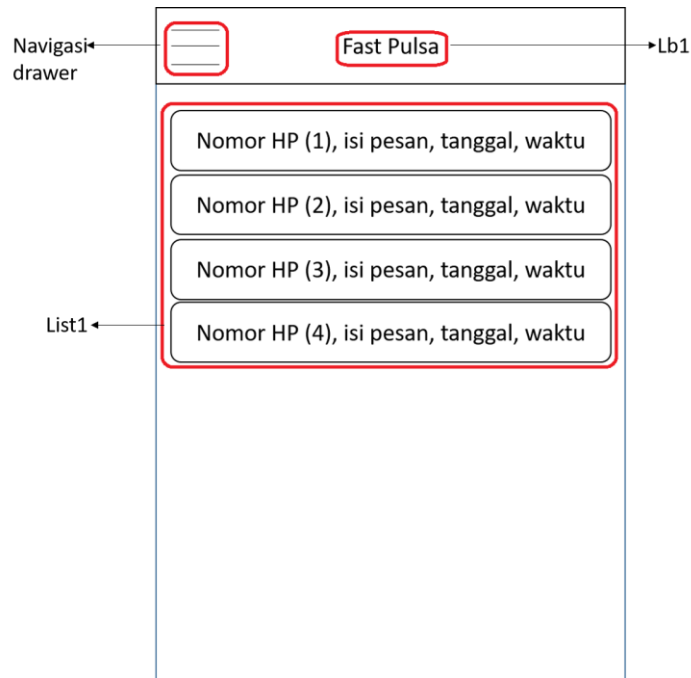
Tabel 3. 28 Perancangan antarmuka pengaturan kata sandi

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Btn2	Button	Reset	Jika diklik maka data-data yang sebelumnya telah diisi akan terhapus dan edit text akan kembali kosong
Btn3	Button	Simpan	Jika diklik maka kata sandi aplikasi yang telah diatur akan tersimpan pada basis data aplikasi
Tx1	edit text	Kata sandi aplikasi lama	Tempat untuk memasukkan kata sandi aplikasi yang lama
Tx2	edit text	Kata sandi aplikasi baru	Tempat untuk memasukkan kata sandi aplikasi baru

Id_objek	Jenis	Nama	Keterangan
Tx3	editText	Konfirmasi kata sandi aplikasi baru	Tempat untuk memasukkan konfirmasi kata sandi aplikasi yang baru

Disajikan oleh Tabel 3.28 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka pengaturan kata sandi.

3.10.12 Perancangan antarmuka pesan masuk



Gambar 3. 48 Perancangan antarmuka pesan masuk

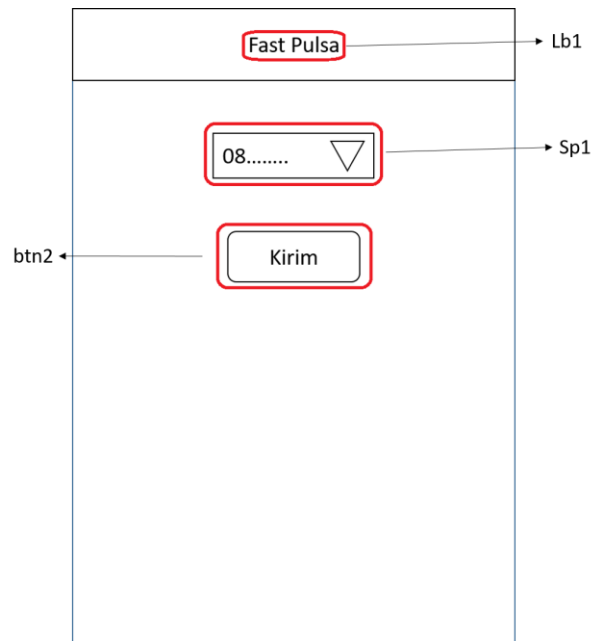
Gambar 3.48 merupakan gambar perancangan antarmuka pesan masuk yang menunjukkan pesan-pesan SMS dari server penyedia pulsa yang ada pada ponsel penjual.

Tabel 3. 29 Deskripsi perancangan antarmuka pesan masuk

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Navigasi drawer	Navigasi drawer	Button navigation drawer	Jika diklik maka akan menampilkan menu utama yang ditampilkan dengan navigation drawer
List1	list view	Pesan masuk	Menampilkan pesan-pesan SMS yang masuk ke ponsel penjual

Disajikan oleh Tabel 3.29 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka pesan masuk.

3.10.13 Perancangan antarmuka cek saldo



Gambar 3. 49 Perancangan antarmuka cek saldo

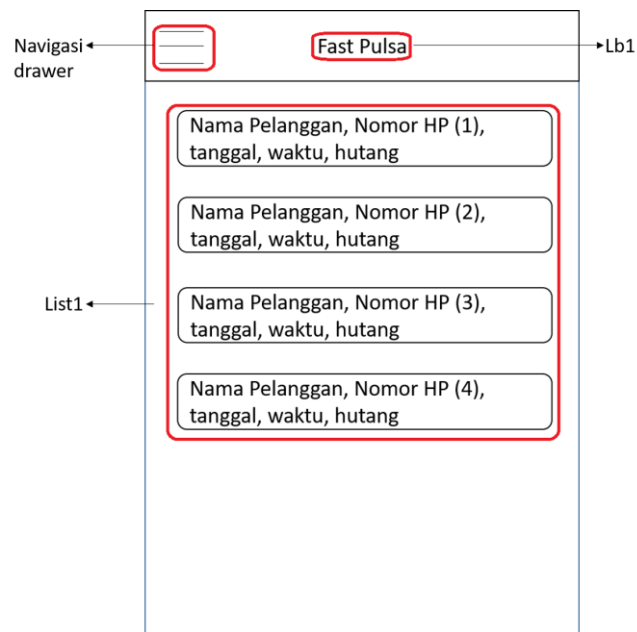
Gambar 3.49 merupakan gambar perancangan antarmuka cek saldo.

Tabel 3. 30 Deskripsi perancangan antarmuka cek saldo

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Btn2	Button	Kirim	Jika diklik maka aplikasi akan memproses permintaan cek saldo penjual ke server penyedia pulsa
Sp1	Spinner	Nomor SMS Center	Menampilkan nomor-nomor SMS Center yang terdaftar pada aplikasi

Disajikan oleh Tabel 3.30 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka cek saldo.

3.10.14 Perancangan antarmuka data hutang



Gambar 3. 50 Perancangan antarmuka data hutang

Gambar 3.50 merupakan gambar perancangan antarmuka data hutang. Berisi data pelanggan yang belum melunasi pembayaran pulsanya. Selain itu, penjual dapat mengubah status pembayaran pelanggan yang telah melunasi pembayaran hutangnya dihalaman ini.

Tabel 3. 31 Deskripsi perancangan antarmuka data hutang

Id_objek	Jenis	Nama	Keterangan
Lb1	Label	Fast pulsa	Nama aplikasi yang menunjukkan aplikasi yang diakses adalah aplikasi fast pulsa
Navigasi drawer	Navigasi drawer	Button navigation drawer	Jika diklik maka akan menampilkan menu utama yang ditampilkan dengan navigation drawer
List1	list view	Data hutang	Menampilkan data pembeli yang belum melunasi pembayaran pulsanya

Disajikan oleh Tabel 3.31 adalah objek-objek beserta id, jenis, nama, dan keterangan dari komponen-komponen antarmuka data hutang.

3.11 Perancangan Algoritma

Perhitungan Total Pendapatan Penjual Pulsa

{Mencari total pendapatan penjual pulsa dari hasil transaksi penjualan pulsa pada bulan tertentu}.

Kondisi awal : penjual telah melakukan transaksi penjualan pulsa

Kondisi akhir : total pendapatan penjual telah didapatkan
 Deskripsi : perhitungan total pendapatan penjual pulsa
 Deklarasi : bulan : string
 total : int

Algoritma :

1. Mengambil nilai harga pada tabel operator dan tgltransaksi pada tabel transaksi.
2. if (totalPendapatan == "SEMUA") then
 total ← Select sum(o.harga) from operator o INNER JOIN
 transaksi t ON o.idOp = t.idOp and t.statuspembayaran =
 'LUNAS'
 else
 total ← Select sum(o.harga) from operator o INNER JOIN
 transaksi t ON o.idOp = t.idOp and t.statuspembayaran =
 'LUNAS' and **strftime('%m', t.tglTransaksi) = bulan**

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Struktur Tabel Basis Data

Adapun data untuk Aplikasi Penjualan Pulsa Elektronik Berbasis Android ini disimpan di dalam basis data SQLite dan juga di dalam file teks. Tabel yang disimpan ke dalam basis data SQLite adalah sebagai berikut:

Nama tabel: Operator

Tabel 4. 1 Implementasi struktur tabel basis data operator

Nama Field	Tipe	Kunci	Atribut
IdOp	Int	PK	Not Null
KodePrefix	Text		Null
NamaOp	Text		Null
Nominal	Int		Null
Harga	Int		Null

Tabel 4.1 merupakan implementasi struktur tabel operator dari perencanaan tabel operator.

Nama tabel: transaksi

Tabel 4. 2 Implemetasi struktur tabel basis data transaksi

Nama Field	Tipe	Kunci	Atribut
idTransaksi	Int	PK	Not Null
IdOp	Int	FK	Not Null
NoHp	Text		Null
StatusPembayaran	Text		Null
TglTransaksi	Date		Null
Waktu	Time		Null
NamaPelanggan	Text		Null

Tabel 4.2 merupakan implementasi struktur tabel transaksi dari perencanaan tabel transaksi.

Nama tabel: smscenter

Tabel 4. 3 Implemetasi struktur tabel basis data smscenter

Nama Field	Tipe	Kunci	Atribut
id	Int	PK	Not Null
nomor	Text		Null

Tabel 4.3 merupakan implementasi struktur tabel smscenter dari perencanaan tabel smscenter.

Sedangkan data yang disimpan dalam bentuk file teks adalah sebagai berikut:

Tabel 4. 4 Implementasi peyimpanan di dalam file teks

Nama File	Isi File
Pengguna.txt	Kata Sandi Aplikasi
PinTransaksi.txt	PIN transaksi pengiriman pulsa
karakterPemisah.txt	Karakter pemisah yang digunakan di dalam format transaksi <i>request</i> pengisian pulsa

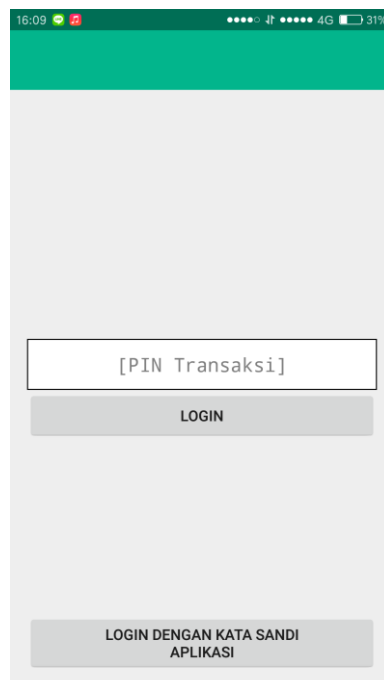
Tabel 4.4 merupakan implementasi struktur tabel pengguna, karakterPemisah, dan PinTransaksi yang di simpan di dalam bentuk file teks.

4.2 Implementasi Proses

4.2.1 Implementasi proses login



Gambar 4. 1 Implementasi antarmuka login dengan kata sandi



Gambar 4. 2 Implementasi antarmuka login dengan PIN transaksi

Gambar 4.1 dan Gambar 4.2. menunjukkan implementasi antarmuka login yang muncul pada saat pertama kali membuka aplikasi dimana penjual pulsa diharuskan

memasukkan kata sandi aplikasi atau memasukkan PIN transaksi yang telah terdaftar ke tempat yang disediakan agar mendapatkan hak akses terhadap aplikasi.

Tabel 4. 5 Tabel file sumber proses login

Nama File	Deskripsi
activity_login.xml	File yang digunakan untuk mengatur <i>layout</i> halaman login
loginKataSandi.java	File yang digunakan untuk menangani inputan login dengan menggunakan kata sandi aplikasi
loginPin.java	File yang digunakan untuk menangani inputan login dengan menggunakan PIN transaksi
ControllerLogin.java	File yang digunakan untuk memverifikasi dan memvalidasi inputan penjual pulsa

Tabel 4.5 menunjukkan file-file yang digunakan untuk proses login.

```
private void validasiLogin() {
    final String bacaKataSandi = cl.bacaFileKataSandi(getApplicationContext());
    if (tx1.getText().toString().equals(bacaKataSandi)) {
        Intent i = new Intent(getApplicationContext(), MainActivity.class);
        startActivity(i);
    } else {
        tx1.setError("Kata Sandi Salah");
        tx1.setText("");
        tx1.requestFocus();
    }
    tx1.setText("");
}
```

Gambar 4. 3 Source code proses login dengan kata sandi yang terdapat pada file loginKataSandi.java

```
private void validasiLogin() {
    final String bacaPin = cl.bacaFilePin(getApplicationContext());
    if (tx1.getText().toString().equals(bacaPin)) {
        Intent i = new Intent(getApplicationContext(), MainActivity.class);
        startActivity(i);
    } else {
        tx1.setError("PIN Transaksi Salah");
        tx1.setText("");
        tx1.requestFocus();
    }
    tx1.setText("");
}
```

Gambar 4. 4 Source code proses login dengan PIN transaksi yang terdapat pada file loginPin.java

Gambar 4.3 menunjukkan implementasi dari *source code* login dengan menggunakan kata sandi sedangkan Gambar 4.4 merupakan implementasi *source code* login dengan menggunakan PIN transaksi dimana ketika penjual mengklik tombol login, aplikasi akan memvalidasi masukan penjual apakah benar atau salah.

Jika benar maka aplikasi akan menampilkan halaman transaksi sedangkan jika salah maka akan muncul pesan kesalahan.

4.2.2 Implementasi proses transaksi pengiriman pulsa

The screenshot shows a mobile application interface for 'FAST PULSA'. At the top, there is a green header with a menu icon and the text 'FAST PULSA'. Below the header, there are several input fields: a phone number field labeled '[Nomor Telepon]' with a '+' button to its right, a name field labeled '[Nama Pembeli]', and a 'Jenis Pulsa' section with a dropdown menu currently showing 'Telkomsel' and '5000'. Below that is an 'SMS Center' field showing '082386436404' and a 'Status Pembayaran' dropdown menu showing 'LUNAS'. At the bottom of the form, there are two green buttons: 'RESET' and 'KIRIM'. The status bar at the top shows the time as 16:09, signal strength, 4G, and 31% battery.

Gambar 4. 5 Implementasi antarmuka transaksi

Gambar 4.5 menunjukkan implementasi antarmuka transaksi yang digunakan untuk menginputkan data-data transaksi pengisian pulsa pelanggan.

Tabel 4. 6 Tabel file sumber proses transaksi

Nama File	Deskripsi
Fragment_transaksi.xml	File yang digunakan untuk mengatur <i>layout</i> halaman transaksi
Transaksi.java	File yang digunakan untuk menangani atribut yang digunakan pada saat proses transaksi
ControllerTransaksi.java	File yang digunakan untuk menangani proses transaksi dan menyimpan riwayat transaksi
TransaksiFragment.java	File yang digunakan untuk menangani inputan penjual pulsa pada form transaksi

Tabel 4.6 menunjukkan file-file yang digunakan pada proses transaksi *request* pengisian pulsa pelanggan.

```

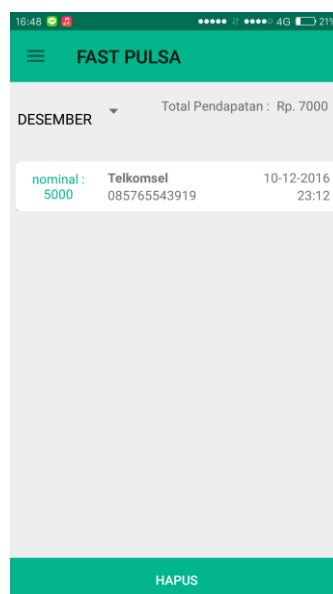
final String idOp = ct.getIdOpFromDB>NamaOp, nom);
final String msg = ct.getKodePrefixFromDB>NamaOp) + ambilNominal(s2.getSelectedItem().toString())
    + ct.bacaKp(mActivity) + phoneTxt.getText().toString()
    + ct.bacaKp(mActivity) + ct.bacaPin(mActivity);
if (phoneTxt.getText().toString().isEmpty()) {
    phoneTxt.setError("Masukkan nomor tujuan");
} else if (s4.getSelectedItem().toString().equals("BELUM LUNAS") && phoneName.getText().toString().isEmpty()) {
    phoneName.setError("Masukkan nama pembeli");
} else if (ct.bacaKp(mActivity).isEmpty()) {
    Toast.makeText(mActivity, "Karakter Pemisah Belum Diatur",
        Toast.LENGTH_SHORT).show();
} else if (ct.bacaPin(mActivity).isEmpty()) {
    Toast.makeText(mActivity, "PIN Transaksi Belum Tersimpan",
        Toast.LENGTH_SHORT).show();
} else {
    final AlertDialog.Builder builder = new AlertDialog.Builder(getContext());
    builder.setMessage("Nomor tujuan      : " + phoneTxt.getText().toString() + "\nNominal Pulsa      : "
        + s2.getSelectedItem().toString() + "\nNomor SMS Center : " + s3.getSelectedItem().toString() +
        "\n\nFormat Transaksi:\n" + msg);
    builder.setPositiveButton("Kirim", (dialogInterface, i) -> {
        ct.insert(new Transaksi(idOp, phoneTxt.getText().toString(), strdate, strtime, stts,
            phoneName.getText().toString());
        KirimSMS(nmr, msg);
        Toast.makeText(mActivity, "Data Berhasil Disimpan",
            Toast.LENGTH_SHORT).show();
        phoneTxt.setText("");
        phoneName.setText("");
    });
    builder.setNegativeButton("TIDAK", (dialogInterface, i) -> {
        //CANCEL
    });
}
}

```

Gambar 4. 6 Source code proses transaksi pada file TransaksiFragment.java

Gambar 4.6 menunjukkan implementasi *source code* untuk menangani proses *request* transaksi pengisian pulsa dimana sebelum proses transaksi dijalankan, aplikasi akan memvalidasi masukan penjual. Jika masukan sudah sesuai, maka aplikasi akan mengirimkan SMS *request* pengisian pulsa pelanggan ke server penyedia pulsa dan data transaksi akan tersimpan.

4.2.3 Implementasi proses pengelolaan riwayat transaksi



Gambar 4. 7 Implementasi antarmuka riwayat transaksi

Gambar 4.7 menunjukkan implementasi antarmuka riwayat transaksi yang menyajikan data-data riwayat transaksi yang ditampilkan dalam bentuk daftar dan total pendapatan yang diperoleh oleh penjual berdasarkan waktu yang diinputkan penjual pulsa, disini pengguna dapat menghapus data riwayat pada bulan tertentu.

Tabel 4. 7 Tabel file sumber proses riwayat transaksi

Nama File	Deskripsi
Fragment_riwayat.xml	File yang digunakan untuk mengatur layout halaman riwayat
Riwayat_row.xml	File yang digunakan untuk mengatur letak komponen di dalam daftar riwayat transaksi
riwayatAdapter.java	File yang digunakan untuk menghubungkan objek data riwayat dari basis data ke dalam daftar riwayat transaksi
Riwayat.java	File yang digunakan untuk menangani atribut yang digunakan pada saat proses riwayat transaksi
ControllerRiwayat.java	File yang digunakan untuk menangani proses riwayat transaksi
RiwayatFragment.java	File yang digunakan untuk menangani antarmuka riwayat transaksi

Tabel 4.7 menunjukkan file-file yang digunakan di dalam proses riwayat transaksi.

```

public List<Riwayat> getAll() {
    List<Riwayat> riwayatList = new ArrayList<>();
    try {
        String selectQuery = "SELECT t.idTransaksi, o.nominal, o>NamaOp, t.noHp, strftime('%d-%m-%Y', t.tglTransaksi) " +
            " as tglTransaksi, t.waktu from operator o INNER JOIN transaksi t ON " +
            "t.statusPembayaran = 'LUNAS' and t.idOp = o.idOp " + "order by t.IdTransaksi DESC";
        open();
        Cursor cursor = database.rawQuery(selectQuery, null);

        if (cursor.moveToFirst()) {
            do {
                Riwayat r = new Riwayat(cursor.getInt(0), cursor.getInt(1), cursor.getString(2), cursor.getString(3),
                    cursor.getString(4), cursor.getString(5));
                riwayatList.add(r);
            } while (cursor.moveToNext());
        }
    } catch (Exception e) {
        Log.v("Info", "ERROR: " + e);
    }
    return riwayatList;
}

public List<Riwayat> getDataBulanan(String bln) {
    List<Riwayat> riwayatList = new ArrayList<>();
    try {
        String selectQuery = "SELECT t.idTransaksi, o.nominal, o>NamaOp, t.noHp, strftime('%d-%m-%Y', t.tglTransaksi) " +
            " as tglTransaksi, t.waktu from operator o INNER JOIN transaksi t ON " +
            "t.statusPembayaran = 'LUNAS' and t.idOp = o.idOp and strftime('%m', t.tglTransaksi) = '" + bln
            + "' order by t.IdTransaksi DESC";
        open();
        Cursor cursor = database.rawQuery(selectQuery, null);

        if (cursor.moveToFirst()) {
            do {
                Riwayat r = new Riwayat(cursor.getInt(0), cursor.getInt(1), cursor.getString(2), cursor.getString(3),
                    cursor.getString(4), cursor.getString(5));
                riwayatList.add(r);
            } while (cursor.moveToNext());
        }
    } catch (Exception e) {
        Log.v("Info", "ERROR: " + e);
    }
    return riwayatList;
}

```

```

public int totalPendapatan() {
    database = dbHelper.getReadableDatabase();
    int total = 0;
    open();
    Cursor c = database.rawQuery("Select sum(o.harga) from operator o INNER JOIN transaksi t ON o.idOp = t.idOp and " +
        "t.statuspembayaran = 'LUNAS'", null);
    c.moveToFirst();
    total = c.getInt(0);
    c.close();
    return total;
}

public int totalPendapatanBulanan(String bln) {
    database = dbHelper.getReadableDatabase();
    int total = 0;
    open();
    Cursor c = database.rawQuery("Select sum(o.harga) from operator o INNER JOIN transaksi t ON o.idOp = t.idOp and " +
        "t.statuspembayaran = 'LUNAS' and strftime('%m', t.tglTransaksi) = '"+bln+"'", null);
    c.moveToFirst();
    total = c.getInt(0);
    c.close();
    return total;
}

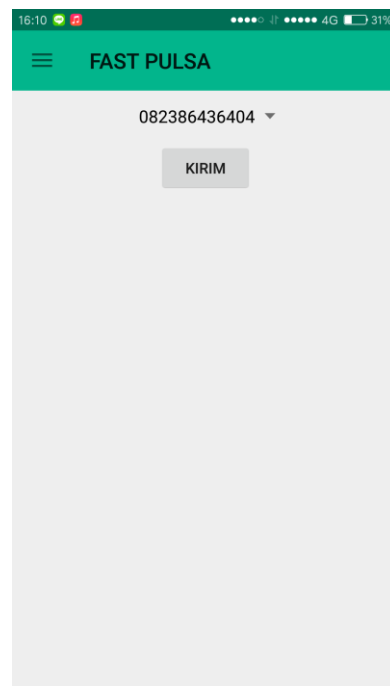
public void hapusDataBulanan(String bln) {
    open();
    database.delete(DBHelper.TABLE_TRANSAKSI, "strftime('%m', tglTransaksi) = '"+bln+"'", null);
    close();
}

```

Gambar 4. 8 Source code proses pengelolaan riwayat transaksi pada file ControllerRiwayat.java

Gambar 4.8 menunjukkan implementasi *source code* di dalam proses pengelolaan riwayat transaksi dimana aplikasi akan menampilkan riwayat transaksi, total pendapatan, dan dapat menghapus riwayat transaksi pada bulan tertentu sesuai dengan yang diminta oleh penjual pulsa.

4.2.4 Implementasi proses *request* cek saldo



Gambar 4. 9 Implementasi antarmuka cek saldo

Gambar 4.9 menunjukkan implementasi antarmuka cek saldo yang digunakan untuk memilih nomor SMS center dan mengirimkan SMS permintaan cek saldo ke server penyedia pulsa.

Tabel 4. 8 Tabel file sumber proses *request* cek saldo

Nama File	Deskripsi
Fragment_ceksaldo.xml	File yang digunakan untuk mengatur layout halaman cek saldo
ControllerCekSaldo.java	File yang digunakan untuk mengani proses pengecekan saldo
CekSaldoFragment.java	File yang digunakan untuk menangani antarmuka pengecekan saldo

Tabel 4.8 menunjukkan file-file yang digunakan dalam proses *request* pengecekan saldo penjual pulsa.

```
private void KirimSMS(String phoneNumber, String message) {
    String SENT = "SMS_SENT";
    String DELIVERED = "SMS_DELIVERED";

    PendingIntent sentPI = PendingIntent.getBroadcast(getContext(), 0,
        new Intent(SENT), 0);

    PendingIntent deliveredPI = PendingIntent.getBroadcast(getContext(), 0,
        new Intent(DELIVERED), 0);

    //---when the SMS has been sent---
    getActivity().registerReceiver((arg0, arg1) → {
        switch (getResultCode()) {
            case Activity.RESULT_OK:
                Toast.makeText(getActivity(), "SMS sent",
                    Toast.LENGTH_SHORT).show();
                break;
            case SmsManager.RESULT_ERROR_GENERIC_FAILURE:
                Toast.makeText(getActivity(), "Generic failure",
                    Toast.LENGTH_SHORT).show();
                break;
            case SmsManager.RESULT_ERROR_NO_SERVICE:
                Toast.makeText(getActivity(), "No service",
                    Toast.LENGTH_SHORT).show();
                break;
            case SmsManager.RESULT_ERROR_NULL_PDU:
                Toast.makeText(getActivity(), "Null PDU",
                    Toast.LENGTH_SHORT).show();
                break;
            case SmsManager.RESULT_ERROR_RADIO_OFF:
                Toast.makeText(getActivity(), "Radio off",
                    Toast.LENGTH_SHORT).show();
        }
    });
}
```

```

        break;
    }
}, new IntentFilter(SENT));

//---when the SMS has been delivered---
getActivity().registerReceiver((arg0, arg1) → {
    switch (getResultCode()) {
        case Activity.RESULT_OK:
            Toast.makeText(getActivity(), "SMS delivered",
                Toast.LENGTH_SHORT).show();
            break;
        case Activity.RESULT_CANCELED:
            Toast.makeText(getActivity(), "SMS not delivered",
                Toast.LENGTH_SHORT).show();
            break;
    }
}, new IntentFilter(DELIVERED));

SmsManager sms = SmsManager.getDefault();
sms.sendTextMessage(phoneNumber, null, message, sentPI, deliveredPI);
}

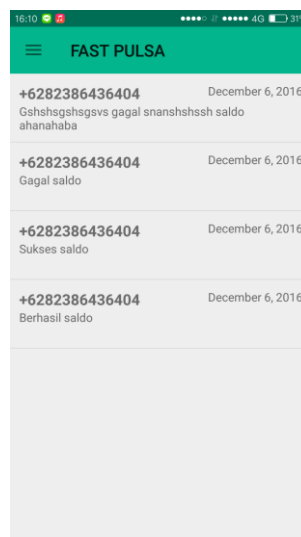
btnKirim = (Button) rootView.findViewById(R.id.btnKirim);
final String msg = "S."+cc.bacaPin(getContext());
btnKirim.setOnClickListener((v) → {
    KirimSMS(sp1.getSelectedItem().toString(), msg);
});

```

Gambar 4. 10 Source code proses request cek saldo pada file CekSaldoFragment.java

Gambar 4.10 menunjukkan implementasi *source code* proses *request* cek saldo yang akan mengirimkan SMS permintaan cek saldo ke server penyedia pulsa berdasarkan nomor SMS center yang telah dipilih oleh penjual pulsa.

4.2.5 Implementasi proses pengecekan inbox



Gambar 4. 11 Implementasi antarmuka inbox

Gambar 4.11 menunjukkan implementasi antarmuka inbox yang menyajikan SMS-SMS yang masuk ke ponsel penjual pulsa dari server pulsa dan disajikan dalam bentuk daftar.

Tabel 4. 9 Tabel file sumber proses pengecekan inbox

Nama File	Deskripsi
Fragment_inbox.xml	File yang digunakan untuk mengatur layout halaman inbox
Pesan_row.xml	File yang digunakan untuk mengatur letak komponen di dalam daftar inbox
InboxFragment.java	File yang digunakan untuk menangani tampilan daftar inbox

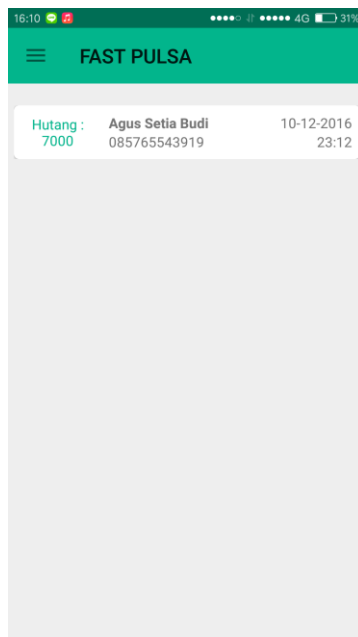
Tabel 4.9 menunjukkan file-file yang digunakan di dalam proses pengecekan inbox.

```
public void displayListView() {
    Uri uriSMS = Uri.parse("content://sms/inbox");
    Cursor cursor = getActivity().getContentResolver().query(uriSMS, null, " (body like '%SUKSES%' or body like '%BERHASIL%' +
        " or body like '%GAGAL%' ) and body like '%SALDO%'", null, null);
    String[] columns = new String[]{"address", "body", "date"};
```

Gambar 4. 12 Source code proses pengecekan inbox pada file InboxFragment.java

Gambar 4.12 menunjukkan implementasi *source code* pada proses pengecekan inbox yang mengambil SMS-SMS yang masuk berdasarkan pola SMS tertentu dari server penyedia pulsa.

4.2.6 Implementasi proses pengelolaan data hutang



Gambar 4. 13 Implementasi antarmuka data hutang

Gambar 4.13 menunjukkan implementasi antarmuka hutang yang menyajikan daftar pelanggan yang belum melunasi pembayaran dan jika mereka sudah melunasinya, penjual pulsa dapat mengubah status pembayaran pelanggan tersebut.

Tabel 4. 10 Tabel file sumber proses pengelolaan data hutang

Nama File	Deskripsi
Fragment_hutang.xml	File yang digunakan untuk mengatur layout halaman data hutang
Hutang_row.xml	File yang digunakan untuk mengatur letak komponen di dalam daftar hutang
Hutang.java	File yang digunakan untuk menangani atribut yang digunakan pada saat proses pengelolaan data hutang
dataHutangAdapter.java	objek data hutang dari basis data ke dalam daftar hutang
ControllerHutang.java	File yang digunakan untuk menangani proses pengelolaan data hutang
HutangFragment.java	File yang digunakan untuk menangani antarmuka data hutang

Tabel 4.10 menunjukkan file-file yang digunakan pada saat proses pengelolaan data hutang.

```

public List<Hutang> getAll() {
    List<Hutang> hutangList = new ArrayList<>();
    try {
        String selectQuery = "SELECT t.idTransaksi, o.harga, t>NamaPelanggan, t.noHp, strftime('%d-%m-%Y', t.tglTransaksi)" +
            " as tglTransaksi, t.waktu from operator o INNER JOIN transaksi t ON t.statusPembayaran = " +
            "'BELUM LUNAS' and t.idOp = o.idOp order by t.IdTransaksi DESC";
        open();
        Cursor cursor = database.rawQuery(selectQuery, null);

        if (cursor.moveToFirst()) {
            do {
                Hutang h = new Hutang(cursor.getInt(0), cursor.getInt(1), cursor.getString(2), cursor.getString(3),
                    cursor.getString(4), cursor.getString(5));
                hutangList.add(h);
            } while (cursor.moveToNext());
        }
    } catch (Exception e) {
        Log.v("Info", "ERROR: " + e);
    }
    return hutangList;
}

public void update (int id){
    open();
    String sql = "Update transaksi set statuspembayaran = 'LUNAS' where idTransaksi = "+id;
    database.execSQL(sql);
}

```

Gambar 4. 14 Source code proses pengelolaan data hutang pada file ControllerHutang.java

Gambar 4.14 menunjukkan implementasi *source code* dari proses pengelolaan data hutang dimana aplikasi akan menampilkan seluruh data pelanggan yang belum melunasi pembayaran, dan penjual pulsa dapat memperbaharui data pelanggan mana yang sudah melunasi pembayaran.

4.2.7 Implementasi proses pengaturan operator



Gambar 4. 15 Implementasi antarmuka pengaturan operator

Gambar 4.15 menunjukkan implementasi antarmuka pengaturan operator yang digunakan untuk menginputkan data kode prefix tiap operator dan harga untuk setiap nominal pembelian pulsa. Data operator juga bisa diperbaharui oleh penjual pulsa.

Tabel 4. 11 Tabel file sumber proses pengaturan operator

Nama File	Deskripsi
Pengaturanoperator.xml	File yang digunakan untuk mengatur layout halaman pengaturan data operator
Operator.java	File yang digunakan untuk menangani atribut yang digunakan pada saat proses pengaturan data operator
ControllerOperator.java	File yang digunakan untuk menangani proses pengaturan data operator
pengaturanOp.java	File yang digunakan untuk menangani antarmuka data operator

Tabel 4.11 menunjukkan file-file yang digunakan pada saat proses pengaturan data operator.

```

//method untuk create/insert operator ke database
public void insert(Operator op) {

    // membuat sebuah ContentValues, yang berfungsi
    // untuk memasangkan data dengan nama-nama
    // kolom pada database

    ContentValues values = new ContentValues();
    values.put(DBHelper.COLUMN_KODEPREFIX, op.getKodePrefix());
    values.put(DBHelper.COLUMN_NAMAOP, op.getNamaOp());
    values.put(DBHelper.COLUMN_NOMINAL, op.getNominal());
    values.put(DBHelper.COLUMN_HARGA, op.getHarga());

    // mengeksekusi perintah SQL insert data
    // yang akan mengembalikan sebuah insert ID
    database.insert(DBHelper.TABLE_OPERATOR, null,
        values);
}

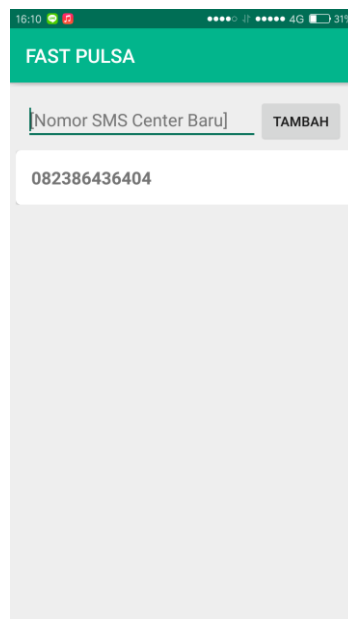
public void update (Operator op){
    open();
    ContentValues values = new ContentValues();
    values.put(DBHelper.COLUMN_KODEPREFIX, op.getKodePrefix());
    values.put(DBHelper.COLUMN_NAMAOP, op.getNamaOp());
    values.put(DBHelper.COLUMN_NOMINAL, op.getNominal());
    values.put(DBHelper.COLUMN_HARGA, op.getHarga());
    database.update(DBHelper.TABLE_OPERATOR, values, DBHelper.COLUMN_IDOP+ " = "+op.getIdPulsa(),
        null);
}

```

Gambar 4. 16 Source code proses pengaturan operator pada file ControllerOperator.java

Gambar 4.16 menunjukkan implementasi *source code* pada proses pengaturan data operator yang menangani inputan penjual pulsa dan akan menyimpan atau memperbaharui data operator.

4.2.8 Implementasi proses pengaturan nomor SMS center



Gambar 4. 17 Implementasi antarmuka pengaturan SMS center

Gambar 4.17 menunjukkan implementasi antarmuka pengaturan nomor SMS center digunakan untuk menyimpan atau menghapus nomor SMS center.

Tabel 4. 12 Tabel file sumber proses pengaturan nomor SMS center

Nama File	Deskripsi
Pengaturansmscenter.xml	File yang digunakan untuk mengatur layout halaman pengaturan SMS center
smscenter_row.xml	File yang digunakan untuk mengatur letak komponen di dalam daftar SMS center
smsCenterAdapter.java	File yang digunakan untuk menghubungkan objek data SMS center dari basis data ke dalam daftar SMS center
SmsCenter.java	File yang digunakan untuk menangani atribut yang digunakan pada saat proses pengaturan SMS center
ControllerSMSCenter.java	File yang digunakan untuk menangani proses pengaturan SMS center
PengaturanSMSCenter.java	File yang digunakan untuk menangani antarmuka pengaturan SMS center

Tabel 4.12 menunjukkan file-file yang digunakan pada saat proses pengaturan nomor SMS center.

```
// FUNGSI UNTUK TAMBAH DATA SMS CENTER
public void tambahNomor(ContentValues values){
    open();
    database.insert(DBHelper.TABLE_SMSCENTER, null, values);
}

// FUNGSI UNTUK AMBIL SEMUA DATA SMS CENTER
public List<SmsCenter> getAll(){
    List<SmsCenter> smsCenterList = new ArrayList<>();
    String selectQuery = "SELECT * FROM " + DBHelper.TABLE_SMSCENTER;

    open();
    Cursor cursor = database.rawQuery(selectQuery, null);

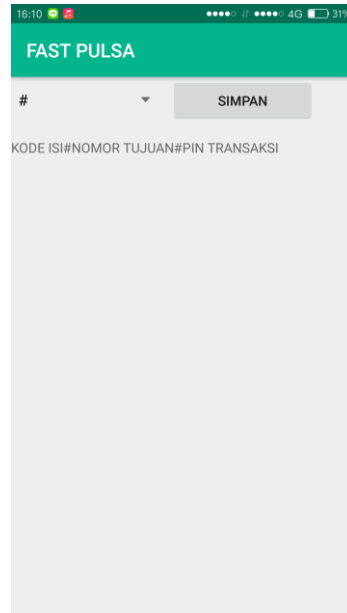
    if (cursor.moveToFirst()){
        do {
            SmsCenter sc = new SmsCenter(cursor.getInt(0), cursor.getString(1));
            smsCenterList.add(sc);
        } while (cursor.moveToNext());
    }
    return smsCenterList;
}

// FUNGSI HAPUS DATA 1 SMS CENTER
public void hapusDatasmCenter(int id) {
    open();
    database.delete(DBHelper.TABLE_SMSCENTER, DBHelper.COLUMN_ID + "=" + id, null);
    close();
}
```

Gambar 4. 18 Source code proses pengaturan SMS center pada file ControllerSMSCenter.java

Gambar 4.18 menunjukkan implementasi *source code* pada proses pengaturan SMS center yang menangani inputan nomor SMS center baru, menampilkan nomor SMS center tersimpan, dan untuk menghapus nomor SMS center.

4.2.9 Implementasi proses pengaturan karakter pemisah



Gambar 4. 19 Implementasi antarmuka pengaturan karakter pemisah

Gambar 4.19 menunjukkan implementasi antarmuka pengaturan karakter pemisah yang digunakan untuk menyimpan karakter pemisah yang digunakan pada format transaksi pengisian pulsa pelanggan sesuai dengan yang ditetapkan oleh server penyedia pulsa.

Tabel 4. 13 Tabel file sumber proses pengaturan karakter pemisah

Nama File	Deskripsi
Pengaturankarakterpemisah.xml	File yang digunakan untuk mengatur layout halaman pengaturan karakter pemisah
ControllerKarakterPemisah.java	File yang digunakan untuk menangani proses pengaturan karakter pemisah
pengaturanKarakterPemisah.java	File yang digunakan untuk menangani tampilan pengaturan karakter pemisah

Tabel 4.13 menunjukkan file-file yang digunakan pada saat proses pengaturan karakter pemisah.

```

private static final String TAG = pengaturanKarakterPemisah.class.getName();
private static final String FILENAME = "karakterPemisah.txt";
pengaturanKarakterPemisah pkp;

public ControllerKarakterPemisah(pengaturanKarakterPemisah pkp) { this.pkp = pkp; }

public void SimpanFile(String kp, Context c) {
    try {
        OutputStreamWriter outputStreamWriter = new OutputStreamWriter(c.openFileOutput(FILENAME, Context.MODE_PRIVATE));
        outputStreamWriter.write(kp);
        outputStreamWriter.close();
    } catch (IOException e) {
        Log.e(TAG, "File write failed: " + e.toString());
    }
}

```

Gambar 4. 20 *Source code* proses pengaturan karakter pemisah pada file **ControllerKarakterPemisah.java**

Gambar 4.20 menunjukkan implementasi *source code* pada proses pengaturan karakter pemisah yang akan disimpan dalam bentuk file teks.

4.2.10 Implementasi proses pengaturan PIN transaksi



Gambar 4. 21 Implementasi antarmuka pengaturan PIN transaksi

Gambar 4.21 menunjukkan implementasi antarmuka pengaturan PIN transaksi yang digunakan untuk menyimpan dan mengubah PIN transaksi dan akan digunakan untuk login ke aplikasi maupun untuk pengisian pulsa pelanggan.

Tabel 4. 14 Tabel file sumber proses pengaturan PIN transaksi

Nama File	Deskripsi
Pengaturanlogin.xml	File yang digunakan untuk mengatur layout halaman pengaturan login
ControllerPengaturanPin.java	File yang digunakan untuk menangani proses pengaturan PIN transaksi
pengaturanPin.java	File yang digunakan untuk menangani tampilan pengaturan PIN transaksi

Tabel 4.14 menunjukkan file-file yang digunakan pada saat proses pengaturan PIN transaksi.

```
private static final String TAG = pengaturanPin.class.getName();
private static final String FILENAME = "PinTransaksi.txt";
pengaturanPin pp;

public ControllerPengaturanPin(pengaturanPin pp) { this.pp = pp; }

public void SimpanFile(String ks, Context c) {
    try {
        OutputStreamWriter outputStreamWriter = new OutputStreamWriter(c.openFileOutput(FILENAME, Context.MODE_PRIVATE));
        outputStreamWriter.write(ks);
        outputStreamWriter.close();
    } catch (IOException e) {
        Log.e(TAG, "File write failed: " + e.toString());
    }
}
```

Gambar 4. 22 Source code proses pengaturan PIN transaksi pada file ControllerPengaturanPin.java

Gambar 4.22 menunjukkan implementasi *source code* pada proses pengaturan PIN transaksi yang akan disimpan dalam bentuk file teks dan digunakan pada saat proses transaksi pulsa dan bisa juga untuk login ke dalam aplikasi.

4.2.11 Implementasi proses mengubah kata sandi aplikasi



Gambar 4. 23 Implementasi antarmuka pengaturan kata sandi aplikasi

Gambar 4.23 menunjukkan implementasi antarmuka pengaturan kata sandi yang digunakan untuk mengubah kata sandi aplikasi dan akan digunakan untuk login ke dalam aplikasi pulsa.

Tabel 4. 15 Tabel file sumber proses pengaturan kata sandi aplikasi

Nama File	Deskripsi
Pengaturanlogin.xml	File yang digunakan untuk mengatur layout halaman pengaturan login
ControllerPengaturanKataSandi.java	File yang digunakan untuk menangani proses pengaturan kata sandi aplikasi
pengaturanKataSandi.java	File yang digunakan untuk menangani tampilan pengaturan kata sandi aplikasi

Tabel 4.15 menunjukan file-file yang digunakan pada saat proses pengaturan kata sandi aplikasi.

```
private static final String TAG = pengaturanKataSandi.class.getName();
private static final String FILENAME = "Pengguna.txt";
pengaturanKataSandi pks;

public ControllerPengaturanKataSandi(pengaturanKataSandi pks) { this.pks = pks; }

public void SimpanFile(String ks, Context c) {
    try {
        OutputStreamWriter outputStreamWriter = new OutputStreamWriter(c.openFileOutput(FILENAME, Context.MODE_PRIVATE));
        outputStreamWriter.write(ks);
        outputStreamWriter.close();
    } catch (IOException e) {
        Log.e(TAG, "File write failed: " + e.toString());
    }
}
```

Gambar 4. 24 *Source code* proses mengubah kata sandi aplikasi pada file **ControllerPengaturanKataSandi.java**

Gambar 4.24 menunjukkan implementasi *source code* pada proses mengubah kata sandi aplikasi yang akan disimpan dalam bentuk file teks dan digunakan pada saat proses login ke dalam aplikasi.

4.3 Pengujian

Tabel 4. 16 Pengujian

Nama Penguji : Nur Zahрати Janah, M. Sc

Tanggal Pengujian : 16 Desember 2016

Tempat Pengujian : Ruang 702 Politeknik Negeri Batam

No.	Nama Proses	Skenario Pengujian	Data Uji	Target	Validasi
1.	Menyimpan kata sandi aplikasi	1. Kata sandi aplikasi belum tersimpan 2. Penjual memasukkan kata sandi aplikasi baru 3. Penjual menekan tombol “Daftar kata sandi”	Kata sandi aplikasi telah diisi Contoh : fast	Kata sandi aplikasi tersimpan dan masuk ke halaman “Transaksi”	
			Kata sandi aplikasi tidak diisi	Gagal menyimpan kata sandi aplikasi dan muncul pesan kesalahan	
2.	Login dengan kata sandi aplikasi	1. Penjual memasukkan kata sandi aplikasi 2. Penjual mengklik tombol “Login”	Kata Sandi benar Contoh : fast	Login berhasil dan masuk ke halaman “Transaksi”	
			Kata Sandi salah Contoh : login	Login gagal dan muncul pesan kesalahan	
3.	Menyimpan PIN transaksi	1. Penjual memilih menu pengaturan pada halaman menu utama 2. Penjual memilih menu “PIN transaksi”	PIN transaksi baru dan konfirmasi PIN transaksi baru sesuai Contoh : 1234	PIN transaksi baru berhasil disimpan	

No.	Nama Proses	Skenario Pengujian	Data Uji	Target	Validasi
		3. Penjual memasukkan PIN transaksi baru 4. Penjual memasukkan kembali PIN transaksi yang baru 5. penjual mengklik tombol “Simpan”	PIN transaksi baru dan konfirmasi PIN transaksi baru tidak sesuai Contoh : PIN baru : 456 Konfirmasi PIN baru : 1234	PIN transaksi baru gagal disimpan dan muncul pesan kesalahan	
4.	Login dengan PIN transaksi	1. Penjual memasukkan PIN transaksi 2. Penjual mengklik tombol “Login”	PIN transaksi benar Contoh : 1234	Login berhasil dan masuk kehalaman “Transaksi”	
			PIN transaksi salah Contoh : 4321	Login gagal dan muncul pesan kesalahan	
5.	Menambah data operator pulsa baru	1. Penjual memilih menu “pengaturan” 2. Penjual memilih menu “operator” 3. Penjual memilih nama operator 4. Penjual memasukkan kode prefix untuk operator terpilih 5. Penjual memasukkan harga untuk setiap nominal pulsa 6. Penjual mengklik tombol “simpan”	Kode prefix dan harga untuk setiap nominal pulsa terisi	Data operator berhasil disimpan	
			Salah satu dari field kode prefix atau harga untuk setiap nominal pulsa tidak diisi	Penyimpanan data operator gagal dan muncul pesan kesalahan	
6.	Memperbaharui data operator	1. Penjual memilih menu “pengaturan” 2. Penjual memilih menu “operator” 3. Penjual memilih nama operator 4. Penjual mengganti data yang ditampilkan oleh aplikasi 5. Penjual mengklik tombol “simpan”	Kode prefix dan harga untuk setiap nominal pulsa terisi	Data operator berhasil diperbaharui	
			Salah satu dari field kode prefix atau harga untuk setiap nominal pulsa tidak diisi	Pembaharuan data operator gagal dan muncul pesan kesalahan	
7.	Menambah nomor SMS Center	1. Penjual memilih menu “pengaturan”	Nomor SMS Center diisi Contoh : 082386436404	Nomor SMS Center baru berhasil ditambahkan	

No.	Nama Proses	Skenario Pengujian	Data Uji	Target	Validasi
		<ol style="list-style-type: none"> 2. Penjual memilih menu “Nomor SMS Center” 3. Penjual memasukkan nomor SMS Center baru 4. Penjual mengklik tombol “Simpan” 	<p>Nomor SMS Center tidak diisi</p> <p>Nomor SMS center sudah ada pada basis data aplikasi Contoh : 082386436404</p>	<p>Proses menambah data SMS Center gagal dan muncul pesan kesalahan</p> <p>Proses menambah data SMS Center gagal dan muncul pesan kesalahan</p>	
8.	Hapus nomor SMS Center	<ol style="list-style-type: none"> 1. Penjual memilih menu “pengaturan” 2. Penjual memilih menu “Nomor SMS center” 3. Penjual memilih nomor SMS center yang ingin dihapus 4. Muncul pesan konfirmasi 	<p>Penjual memilih “YA”</p> <p>Penjual memilih “TIDAK”</p>	<p>Nomor SMS Center berhasil dihapus</p> <p>Nomor SMS Center tidak dihapus</p>	
9.	Menyimpan karakter pemisah format transaksi	<ol style="list-style-type: none"> 1. Penjual memilih menu pengaturan pada halaman menu utama 2. Penjual memilih menu “Karakter Pemisah” 3. Penjual memilih karakter pemisah format transaksi 4. penjual mengklik tombol “Simpan” 	Karakter pemisah terpilih Contoh : #	Karakter pemisah berhasil disimpan	
10.	Mengubah PIN transaksi	<ol style="list-style-type: none"> 1. Penjual memilih menu pengaturan pada halaman menu utama 2. Penjual memilih menu “PIN transaksi” 3. Penjual memasukkan PIN transaksi lama 4. Penjual memasukkan PIN transaksi baru 	PIN transaksi lama, PIN transaksi baru, dan konfirmasi PIN transaksi baru benar	PIN transaksi baru berhasil disimpan	

No.	Nama Proses	Skenario Pengujian	Data Uji	Target	Validasi
		5. Penjual memasukkan kembali PIN transaksi yang baru 6. penjual mengklik tombol “Simpan”	PIN transaksi lama salah Contoh : 321	PIN transaksi baru gagal disimpan dan muncul pesan kesalahan	
			PIN transaksi lama tidak diisi	PIN transaksi baru gagal disimpan dan muncul pesan kesalahan	
			PIN transaksi baru tidak diisi	PIN transaksi baru gagal disimpan dan muncul pesan kesalahan	
			Konfirmasi PIN transaksi baru tidak diisi	PIN transaksi baru gagal disimpan dan muncul pesan kesalahan	
			PIN transaksi baru dan konfirmasi PIN transaksi baru tidak sesuai Contoh : PIN baru : 456 Konfirmasi PIN baru : 1234	PIN transaksi baru gagal disimpan dan muncul pesan kesalahan	
11.	Mengubah kata sandi aplikasi	1. Penjual memilih menu pengaturan pada halaman menu utama 2. Penjual memilih menu “Kata Sandi”	kata sandi aplikasi baru dan konfirmasi kata sandi aplikasi baru sesuai Contoh : Kata sandi baru : fastpulsa	Kata sandi aplikasi baru berhasil disimpan	

No.	Nama Proses	Skenario Pengujian	Data Uji	Target	Validasi
		3. Penjual memasukkan kata sandi aplikasi baru 4. Penjual memasukkan kembali kata sandi aplikasi yang baru 5. Penjual mengklik tombol “Simpan”	Konfirmasi kata sandi baru : fastpulsa kata sandi aplikasi baru dan konfirmasi kata sandi aplikasi baru tidak sesuai Kata sandi baru : fastpulsa Konfirmasi kata sandi baru : admin	Kata sandi aplikasi baru gagal disimpan dan muncul pesan kesalahan	
12.	Melakukan transaksi pengiriman pulsa	1. Penjual memilih menu “transaksi” 2. Penjual memasukkan nomor hp pelanggan 3. Penjual memilih operator pembelian pulsa 4. Penjual memilih nominal pembelian pulsa 5. Penjual memilih nomor SMS Center yang tersedia 6. Penjual mengkonfirmasi pembayaran telah “Lunas” 7. Muncul pesan konfirmasi	Penjual memilih “Kirim” Penjual memilih “Tidak” Penjual mengkonfirmasi pembayaran “belum lunas” Penjual mengkonfirmasi pembayaran “belum lunas” dan tidak mengisi text field nama pelanggan Data transaksi yang dimasukkan tidak lengkap	SMS permintaan pengisian pulsa dikirimkan, transaksi pulsa tercatat Pengiriman pulsa tidak diproses Muncul pesan kesalahan yang menunjukkan agar mengisi nama pembeli Muncul pesan kesalahan dan Pengiriman pulsa tidak diproses Muncul pesan kesalahan dan Pengiriman pulsa tidak diproses	
13.	Melihat riwayat transaksi dan total pendapatan	1. Penjual memilih menu “riwayat” 2. Penjual memilih periode waktu riwayat transaksi yang ingin dilihat	Data riwayat tersedia Contoh: Riwayat pada bulan desember Data riwayat tidak tersedia	Menampilkan riwayat transaksi dan total pendapatan pada bulan desember Menampilkan form “data tidak ada”	

No.	Nama Proses	Skenario Pengujian	Data Uji	Target	Validasi
14.	Menghapus riwayat transaksi	<ol style="list-style-type: none"> 1. Penjual memilih menu “riwayat” 2. Penjual memilih periode waktu riwayat transaksi yang ingin dihapus 3. Penjual menekan tombol “hapus” 4. muncul pesan konfirmasi 	Penjual memilih “ya”	Data riwayat transaksi yang dipilih telah dihapus	
			Penjual memilih “tidak”	Data riwayat transaksi yang dipilih tidak dihapus	
15.	Cek Saldo	<ol style="list-style-type: none"> 1. Penjual memilih menu “cek saldo” 2. Penjual memilih nomor SMS Center yang tersedia 3. Penjual menekan tombol “Kirim” 		SMS permintaan pengecekan saldo dikirim ke server pulsa	
16.	Cek Inbox	<ol style="list-style-type: none"> 1. Penjual memilih menu “Inbox” 	Data SMS yang mengandung pesan “Berhasil”, “sukses”, “gagal”, “saldo”, dan “stok” tersedia	Menampilkan seluruh SMS yang mengandung pesan “Berhasil”, “sukses”, “gagal”, “saldo”, dan “stok”	
			Data SMS yang mengandung pesan “Berhasil”, “sukses”, “gagal”, “saldo”, dan “stok” tidak tersedia	Menampilkan form “data tidak ada”	
17.	Menghapus Pesan SMS masuk	<ol style="list-style-type: none"> 1. Penjual memilih menu “Inbox” 2. Penjual memilih pesan SMS yang ingin dihapus 3. Penjual menekan tombol “hapus” 4. Muncul pesan konfirmasi 	Penjual memilih “ya”	Pesan SMS yang dipilih telah dihapus	
			Penjual memilih “tidak”	Pesan SMS yang dipilih tidak dihapus	
18.	Mengubah status pembayaran	<ol style="list-style-type: none"> 1. Penjual memilih menu “Hutang” 2. Penjual memilih data pelanggan yang telah melunasi pembayaran 3. Muncul pesan konfirmasi 	Penjual memilih “ya”	Status pembayaran telah diubah	
			Penjual memilih “tidak”	Status pembayaran tidak diubah	

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan tahapan analisis, implementasi, dan hasil pengujian yang telah didapat sebelumnya, maka diperoleh kesimpulan sebagai berikut:

1. Fungsi login pada aplikasi bekerja dengan baik untuk mengamankan data yang ada pada aplikasi sehingga aplikasi tidak dapat diakses oleh sembarang orang, khususnya untuk data laporan hutang.
2. Fitur transaksi pengiriman pulsa telah diuji dan berhasil mengirimkan permintaan pengiriman pulsa kepada server penyedia pulsa sesuai dengan format yang berlaku dan pulsa pelanggan berhasil diisi.
3. Menyediakan laporan riwayat transaksi dan total pendapatan penjual pulsa pada bulan tertentu sehingga bisa digunakan untuk menggantikan sistem transaksi secara manual dengan menulis dibuku catatan penjualan.
4. Fitur hutang telah berhasil digunakan untuk mengelola data pelanggan yang belum melunasi pembayaran.

5.2 Saran

Penulis menyadari masih banyak kekurangan yang ada pada aplikasi penjualan pulsa elektronik ini. Oleh karena itu, untuk meningkatkan kinerja dan kemampuan aplikasi yang dibangun agar menjadi aplikasi yang benar-benar handal, penulis memiliki beberapa saran sebagai berikut :

1. Diharapkan peneliti selanjutnya dapat menambahkan fitur penambahan saldo/stok
2. Transaksi bisa dilakukan dengan menggunakan SIM 1 maupun SIM 2
3. Menambahkan fitur pencarian SMS masuk dan pencarian nama pelanggan hutang
4. Tampilan aplikasi agar dibuat lebih menarik

DAFTAR PUSTAKA

- A.S, Rosa & Shalahuddin, M. (2011). *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung: Modula.
- Fowler, M. (2005). *UML Distilled : Panduan Singkat Bahasa Pemodelan Objek Standar* (3 ed.). Yogyakarta: Andi.
- Henderi. (2007). *Diktat Kuliah UML STMIK AMIKOM*. Bandung: STMIK AMIKOM.
- Purba, A. E., Budi, A. S., & Prasetyo, D. T. (2015). *Aplikasi Penjualan Pulsa Elektronik*. Batam: Politeknik Negeri Batam.
- Safaat H., N. (2011). *Programan Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.
- Setya, I. B. (2015). *Aplikasi Penjualan Pulsa Elektronik Berbasis Android*. Sidoarjo: Universitas Muhammadiyah Sidoarjo.
- Supardi, Y. (2014). *Semua Bisa Menjadi Programmer Android*. Jakarta: PT Elex Media Komputindo.
- Wahana Komputer. (2013). *Shotcourse Series : Android Programming with Eclipse*. Semarang: Penerbit Andi.
- Wibisono, G., Kurniawan Usman, U., & Hantoro, G. D. (2008). *Konsep Teknologi Seluler*. Bandung: Informatika.
- Yulianti, E., & Rusyiatno, T. (2013, April). Rekayasa Perangkat Lunak Penjualan Pulsa Electric Berbasis SMS Menggunakan Visual Basic. *TEKNOIF, 1*.