

## PENDAHULUAN

### 1.1 Latar Belakang

Semakin meningkatnya penggunaan telepon seluler, berdampak besar terhadap proses bisnis jual beli pulsa. Bukan hanya sekedar untuk komunikasi SMS dan telepon, dengan berkembangnya teknologi saat ini, telepon seluler juga digunakan untuk mengakses web yang membutuhkan biaya dalam pemakaiannya. Dengan berbagai fasilitas yang disuguhkan untuk memenuhi kebutuhan pengguna, tentunya hal tersebut semakin menambah jumlah kebutuhan pulsa bagi setiap individu [ALT-10]. Ditambah lagi dengan banyaknya beberapa provider yang menawarkan produk dan paket internet dengan harga yang terjangkau, hal tersebut secara langsung mendukung proses munculnya bisnis jual-beli pulsa sebagai mediasi antara pengguna dengan operator untuk memenuhi kebutuhan terhadap permintaan pulsa. [AR-10]

Counter pulsa adalah agen yang melayani jasa pengisian pulsa. Layanan atau sistem transaksi yang ada pada counter pulsa saat ini sangatlah sederhana. Sebagai contoh, pembeli datang ke counter, menyebutkan nomor telepon, atau hanya menulis pada suatu kertas lalu diserahkan kepada pihak penjual. Hal ini tentunya akan banyak menemui masalah apabila transaksi yang dilakukan bersifat masal. Misalnya dalam 1 kali transaksi ada lebih dari 20 pembeli datang untuk membeli pulsa dengan menulis di kertas nomor telepon yang akan diisi. Bisa saja dalam transaksi tersebut terjadi kesalahan penginputan no tujuan, tertukarnya nominal pulsa yang ingin dikirim, atau bahkan terkirimnya pulsa kepada pembeli lebih dari satu kali jika terjadi masalah jaringan. Tentunya yang rugi dan yang akan disalahkan adalah pihak penjual, belum lagi jika terjadi *error* transaksi yang menyebabkan pulsa tidak masuk pada pembeli. Pembeli tidak bisa melihat status transaksi, karena hal tersebut hanya bisa diketahui oleh pihak penjual saja melalui notifikasi yang diberikan oleh server.

Berdasarkan masalah diatas, maka dibuatlah suatu sistem aplikasi berbasis desktop agar proses transaksi berjalan lebih efektif dan efisien. Aplikasi tersebut menggantikan tampilan serta proses layanan yang selama ini ada. Contohnya saat ini transaksi pada counter-counter pulsa dilakukan secara manual dengan format

sms, dengan adanya aplikasi ini *user* dalam hal ini pembeli bisa lebih mudah dalam proses pembelian pulsa dengan hanya menekan tombol-tombol yang ada pada halaman aplikasi desktop lalu memilih nominal pulsa serta provider apa yang digunakan. Fitur yang disediakan juga memungkinkan pengguna untuk melakukan transaksi lebih dari 1 kali proses. Berbeda dengan sistem sms yang melakukan transaksi dengan memproses transaksi satu per satu. Hal tersebut tentunya kurang efisien mengingat waktu yang dibutuhkan lumayan lama untuk satu kali prosesnya. Sebagai output transaksi, sistem akan menampilkan nomor faktur atau no transaksi dari transaksi yang dilakukan. Nomor faktur tersebut nantinya oleh pembeli diserahkan kepada kasir yang bertugas memasukkan nomor faktur tersebut ke dalam aplikasi kasir pada *work station* dan memprosesnya. Selain itu, fitur-fitur yang ditawarkan pada aplikasi ini yaitu pembeli juga akan mendapatkan notifikasi dengan menerima sms pending mengenai status transaksi apabila terjadi pending atau gangguan pada pengiriman pulsa. Aplikasi ini tentunya semakin mempermudah pekerjaan operator, bukan hanya karena tidak perlu lagi proses sms secara manual, hal ini juga bisa menjawab permasalahan yang selama ini terjadi pada pihak penjual diantaranya kesalahan penginputan nomor, *delay transaction*, dan *management* waktu dalam sekali proses transaksi. Pelayanan terhadap pembeli juga semakin komplit dengan banyaknya fitur-fitur yang membantu pengguna dalam memonitoring transaksi yang telah dilakukan.

Berdasarkan latar belakang yang telah dijabarkan, maka dibuatlah suatu sistem aplikasi dengan judul “Rancang Bangun Aplikasi Desktop dengan Nomor Faktur sebagai Antrian Proses” untuk mengatasi permasalahan yang ada pada transaksi pulsa yang ada saat ini. Sehingga transaksi pulsa yang ada dapat berjalan lebih efektif dan efisien.

## 1.2 Rumusan Masalah

1. Analisis kebutuhan *user* dan administator untuk merancang sistem aplikasi desktop penjualan pulsa dengan nomor faktur sebagai antrian proses
2. Mengembangkan dan mengimplementasikan aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses

3. Menguji sistem aplikasi penjualan pulsa dengan nomor faktur sebagai antrian proses

### 1.3 Batasan Masalah

Batasan dari penelitian ini adalah :

1. Counter pulsa yang dimaksud adalah First Cell sebagai tempat studi kasus dan salah satu server maupun distributor pulsa.
2. Provider yang digunakan pada aplikasi ini adalah voucher yang dijual pada counter "First Cell". Diantaranya adalah sebagai berikut :
  - a. Indosat
  - b. Telkomsel
  - c. Three (3)
  - d. Esia
  - e. XI
  - f. Smartfren
  - g. Flexi
  - h. Axis
3. *Basis Data* yang digunakan adalah basis data *MySQL*.
4. Bahasa Pemrograman yang digunakan pada aplikasi ini adalah bahasa pemrograman delphi menggunakan software Embarcadero RAD2.

### 1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Menganalisis kebutuhan *user* untuk merancang aplikasi desktop penjualan pulsa dengan nomor faktur sebagai antrian proses.
2. Dapat mengimplementasikan aplikasi desktop penjualan pulsa dengan menggunakan *SMS Gateway*.
3. Menguji aplikasi desktop penjualan pulsa dengan nomor faktur sebagai antrian proses untuk meningkatkan kenyamanan *user*.

### 1.5. Manfaat Penelitian

Penelitian ini diharapkan dapat bermanfaat untuk berbagai pihak.

Diantaranya :

#### 1. Bagi penulis

- Mengaplikasikan ilmu yang didapat selama mengikuti perkuliahan di Teknik Informatika Universitas Brawijaya
- Mendapatkan pengetahuan tentang pembuatan aplikasi desktop dengan memanfaatkan aplikasi *SMS Gateway* sebagai notifikasi pending, serta pengolahan database pada sistem dengan output nomor faktur sebagai antrian proses transaksi.

#### 2. Bagi Agen Pulsa (Konter)

- Meminimalisir kesalahan yang dilakukan operator dalam penginputan nomor transaksi karena penginputan nomor transaksi dilakukan sendiri oleh *user* (pembeli).
- Meminimalisir kesalahan operasional yang memungkinkan dilakukan oleh penjual dalam melakukan transaksi.

#### 3. Bagi *User* (Pembeli)

- Pembeli sebagai aktor utama dalam proses transaksi pulsa. Segala aktifitas transaksi dilakukan langsung oleh pembeli.
- Fasilitas notifikasi pending berbasis *SMS Gateway* bermanfaat untuk pembeli sebagai monitoring transaksi yang dilakukan.
- Nomor faktur bermanfaat mengurangi terjadinya kesalahan penginputan transaksi karena nomor faktur tersebut hanya memunculkan transaksi yang disimpan oleh nomor faktur yang bersangkutan

## 1.6 Sistematika Penulisan

### **BAB I Pendahuluan**

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

### **BAB II Dasar Teori**

Membahas teori-teori yang mendukung dalam pengembangan dan perancangan aplikasi penjualan pulsa berbasis desktop dengan nomor faktur sebagai antrian proses

### **BAB III Metodologi Penelitian**

Membahas tentang metode yang digunakan dalam penulisan yang terdiri dari studi literatur, perancangan perangkat lunak, implementasi perangkat lunak, pengujian dan analisis.

### **BAB IV Perancangan dan Implementasi**

Membahas tentang analisa kebutuhan dan perancangan dari aplikasi penjualan pulsa berbasis desktop serta implementasi sistem dengan menerapkan nomor faktur sebagai antrian proses.

### **BAB V Pengujian**

Memuat tentang hasil pengujian dan analisis terhadap sistem yang telah direalisasikan.

### **BAB VI Penutup**

Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian perangkat lunak yang dikembangkan dalam skripsi ini serta saran – saran untuk pengembangan lebih lanjut.

## BAB II

### KAJIAN PUSTAKA DAN DASAR TEORI

Pada bab dua, terdiri dari kajian pustaka dan dasar teori. Kajian pustaka membahas penelitian yang telah ada dan yang diusulkan, yaitu penelitian sebelumnya yang berjudul '*Pembuatan Aplikasi Sistem Penjualan Pulsa Pada Toko Miyo Cell Menggunakan Neatbeans 6.0.1 dan MySQL*' dan penelitian saat ini yaitu '*Rancang Bangun Aplikasi Desktop Sistem Penjualan Pulsa Dengan Nomor Faktur Sebagai Antrian Proses*'. Dasar teori membahas teori yang diperlukan untuk menyusun penelitian yang diusulkan. Pada penelitian ini, dasar teori yang diperlukan adalah konsep dasar Sistem Penjualan Pulsa, konsep dasar *SMS Gateway*, konsep dasar *AT-Command*, pengembangan dan rekayasa perangkat lunak, serta konsep dasar teknik dan strategi pengujian perangkat lunak.

#### 2.1 Kajian Pustaka

Pada penelitian penelitian sebelumnya yang berjudul '*Pembuatan Aplikasi Sistem Penjualan Pulsa Pada Toko Miyo Cell Menggunakan Neatbeans 6.0.1 dan MySQL*', aplikasi digunakan hanya sebagai fitur mengisi pulsa dalam bertransaksi. Pembeli hanya menyebutkan nomor yang akan diisi lalu penjual memproses transaksi. Tidak ada notifikasi untuk *user* (pembeli) terhadap transaksi yang dilakukan.

Aplikasi pada penelitian ini bekerja sesuai dengan masukan dari penjual pulsa. Apabila terjadi kesalahan penginputan nomor (*human error*) oleh penjual, tentu *user* (pembeli) tidak akan mengetahuinya. Terlebih lagi, pada sistem aplikasi tersebut tidak mempunyai fitur notifikasi terhadap transaksi yang dilakukan, misalnya gangguan operator yang menyebabkan transaksi pulsa pending.

Sedangkan pada penelitian yang dilakukan, aplikasi bekerja sesuai dengan masukan dari *user* (pembeli). Segala kesalahan dalam transaksi tentunya akan terlepas dari tanggung jawab penjual. Pada perancangan aplikasi ini terdapat aplikasi tambahan berupa aplikasi kasir, dimana aplikasi tersebut *terintegrasi* dengan aplikasi utama sebagai tampilan penjualan pulsa. Ketika *user* (pembeli) selesai melakukan transaksi, maka *user* akan mendapatkan tampilan nomor faktur berupa nomor pembelian, jumlah tagihan, dan banyaknya transaksi yang

dilakukan. Nomor pembelian tersebut digunakan sebagai tanda telah melakukan transaksi dan admin melakukan konfirmasi dengan menginputkan nomor pembelian terhadap aplikasi kasir dan memproses transaksi tersebut. Aplikasi *SMS Gateway* berfungsi sebagai notifikasi apabila terjadi gangguan operator yang menyebabkan transaksi pending.

Perbedaan pada kedua penelitian tersebut adalah pada tahap data masukan dan proses aplikasi. Pada penelitian awal, data masukan dilakukan oleh penjual sedangkan pembeli hanya bertugas menyebutkan nomor telepon kepada penjual dan memprosesnya. Sedangkan pada penelitian ini, segala aktifitas transaksi dilakukan oleh pembeli itu sendiri. Dimana proses menginputkan nomor, memilih nominal pulsa, dan transaksi lain dilakukan sendiri. Tugas penjual sebagai admin (kasir) hanya mengkonfirmasi transaksi dengan memasukkan nomor transaksi yang di dapat oleh *user* ke aplikasi kasir.

## 2.2 Dasar Teori

Dasar teori membahas teori yang diperlukan untuk menyusun penelitian yang diusulkan. Dasar teori yang digunakan adalah *SMS Gateway*, *AT-Command*, rekayasa perangkat lunak, pengujian perangkat lunak, pengujian UAT (*User Acceptance Testing*).

### 2.2.1 *SMS Gateway*

*Short Message Service* (SMS) adalah protokol layanan pertukaran pesan text singkat (sebanyak 160 karakter per pesan) antar telepon. *SMS* diciptakan sebagai standart pesan oleh ETSI (Europesan Telecommunication Standards Institute), yang juga menmbuat standar GSM yang diimplementasikan oleh semua operator GSM. [TFK-08].

*SMS Gateway* diartikan sebagai suatu platform yang menyediakan mekanisme untuk EUA menghantar dan menerima SMS dari peralatan mobile (HP, PDA, Handphone, dll) melalui *SMS Gateway's shortcode* [AFR-08].

Pada prinsipnya *SMS Gateway* adalah sebuah perangkat lunak yang menggunakan bantuan komputer dan memanfaatkan teknologi seluler yang *diintegrasikan* guna mendistribusikan pesan-pesan yang *di-generate* lewat sistem informasi melalui media SMS yang *di-handle* oleh jaringan seluler [FA-08]

Istilah *gateway* dapat diartikan sebagai pintu gerbang. Namun pada dunia komputer, *gateway* dapat diartikan sebagai jembatan penghubung antara satu sistem dengan sistem yang lain, sehingga dapat terjadi pertukaran data antar sistem tersebut. *SMS Gateway* adalah sebuah aplikasi yang merubah proses *SMS* dari *Mobile-Equipment* ke PC/ Laptop, *SMS* seperti layaknya fitur telepon seluler, tetapi ada perbedaan dari segi fitur, dan fungsi yang bisa di buat berdasarkan kebutuhan bisnis [PUR-10].

Cara kerja *SMS Gateway* : *SMS* dikirim pemilih ke nomor yang telah ditentukan, isi sms merupakan Format yang telah ditentukan sebelumnya. *SMS* akan dikirim melalui *modem pool / handphone* ke nomor yang ditentukan. Setelah itu akan menunggu balasan, jika telah ada balasan akan disimpam pada *database*.

### 2.2.2 AT-Command

Perintah *AT Command* (*Attention Command*) digunakan untuk berkomunikasi dengan terminal melalui *serial port* pada komputer. *AT Command* merupakan inti dari sebuah *SMS gateway* karena berisi perintah-perintah yang memungkinkan sebuah komputer untuk dapat mengetahui kekuatan sinyal dari terminal, mengirim pesan singkat (*SMS*), membaca pesan dan dapat mengetahui nomor *IMEI* (*International Mobile Equipment Identity*). Umumnya *AT Command* digunakan pada *Hyper Terminal*. Dibawah ini merupakan beberapa contoh *AT Command*

**Tabel 2.1 Tabel AT Command**

Perintah	Kegunaan
AT	Mengecek apakah <i>handphone</i> telah terhubung
AT+CMGF	Menetapkan format <i>mode</i> dan terminal
AT+CSCS	Menetapkan jenis <i>encoding</i>
AT+CNMI	Mendeteksi pesan <i>SMS</i> yang baru masuk secara otomatis
AT+CMGL	Membuka daftar <i>SMS</i> yang ada pada

	<i>SIM Card</i>
AT+CMGS	Mengirim Pesan <i>SMS</i>
AT+CMGR	Membaca Pesan <i>SMS</i>
AT+CMGD	Menghapus Pesan <i>SMS</i>

**Sumber : [WIY-11]**

Dalam program SMS Server yang akan kita buat nanti, tidak semua perintah AT digunakan. Kita hanya menggunakan beberapa perintah AT yang ada hubungannya dengan sistem kerja dari program Sms server [PA-07].

### 2.2.3 Rekayasa Perangkat Lunak

Untuk Rekayasa perangkat lunak adalah disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal spesifikasi sistem sampai pemeliharaan sistem setelah digunakan [SOM-03:7]. Pada definisi ini, ada dua istilah kunci :

1. 'disiplin rekayasa'

Perekayasa membuat suatu alat bekerja. Perekayasa menerapkan teori, metode, dan alat bantu yang sesuai. Perekayasa menggunakannya dengan selektif dan selalu mencoba mencari solusi terhadap permasalahan, walaupun tidak ada teori atau metode yang mendukung. Perekayasa juga menyadari bahwa mereka harus bekerja dalam batasan organisasi dan keuangan, sehingga mereka berusaha mencari solusi dalam batasan – batasan ini.

2. 'semua aspek produksi perangkat lunak'

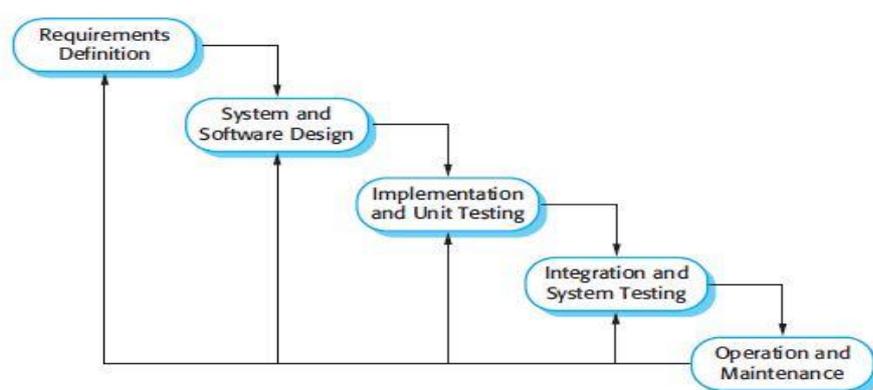
Rekayasa perangkat lunak tidak hanya berhubungan dengan proses teknis dari pengembangan perangkat lunak tetapi juga dengan kegiatan seperti manajemen proyek perangkat lunak dan pengembangan alat bantu, metode dan teori untuk mendukung produksi perangkat lunak.

Perekayasa perangkat lunak memakai pendekatan yang sistematis dan terorganisir terhadap pekerjaan mereka karena cara ini seringkali paling efektif untuk menghasilkan perangkat lunak berkualitas tinggi. Rekayasa ini mencakup masalah pemilihan metode yang paling sesuai untuk satu set keadaan dan

pendekatan yang lebih kreatif, informal terhadap pengembangan yang mungkin efektif pada beberapa keadaan [SOM-03:7].

Seorang *software engineer* atau sekumpulan *software engineer* harus menggabungkan strategi pengembangan perangkat lunak yang meliputi proses, metode, dan alat bantu yang digunakan dalam proses pengembangan. Strategi ini disebut sebagai model proses (*process model*) atau paradigma rekayasa perangkat lunak (*software engineering paradigm*). Model proses untuk rekayasa perangkat lunak dipilih sesuai dengan sifat dari proyek dan aplikasi yang akan dibuat. Terdapat beberapa model proses untuk rekayasa perangkat lunak antara lain *linear sequential model (waterfall)*, *prototyping*, *Rapid Application Development (RAD)*, *incremental model* dan *spiral model* [PRE-01]. Model proses yang digunakan dalam skripsi ini adalah *waterfall model*.

*Waterfall model* kadang disebut siklus hidup klasik atau model air terjun, model sekuensial linier menunjukkan sistematis, berurutan untuk pengembangan perangkat lunak yang dimulai pada tingkat sistem dan kemajuan melalui analisis, desain, coding, pengujian, dan dukungan [PRE-01]. *Waterfall model* memiliki kelebihan proses-prosesnya mudah dipahami, struktur sistem jelas, sebuah fase dijalankan setelah fase sebelumnya selesai. Proses pengembangan ini digunakan karena kebutuhan *user* telah dipahami dan kemungkinan terjadinya perubahan kebutuhan *user* kecil. Proses pengembangan menggunakan model proses *waterfall* ini dapat dilihat dalam Gambar 2.1.



**Gambar 2.1 Waterfall Model**  
Sumber : [SOM-11]

## 2.2.4 Pengujian Perangkat Lunak

Pengujian perangkat lunak merupakan elemen penting dari penjaminan kualitas perangkat lunak dan merepresentasikan tinjauan utama spesifikasi, desain, dan pembuatan kode [PRE-01]. Teknik atau metode perancangan kasus uji yang digunakan adalah *black-box testing*.

### 2.2.4.1 Black-Box Testing

*Black-box testing* atau *behavioral testing* berfokus pada persyaratan fungsional perangkat lunak [PRE-01]. Dengan demikian, pengujian *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk semua program. Pengujian *black-box* merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan.

Pengujian *black-box* berusaha menemukan kesalahan dalam kategori berikut:

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses basis data eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

Pengujian *black-box* cenderung diaplikasikan selama tahap akhir pengujian. Pengujian *black-box* memperhatikan struktur kontrol, maka perhatian berfokus pada domain informasi.

### 2.2.4.2 Pengujian UAT (*User Acceptance Testing*)

Pengujian UAT digunakan untuk mendapatkan konfirmasi atau kepastian melalui *trial* atau *review* dari pihak yang terkait sesuai dengan requirement yang telah disetujui sebelumnya. Tahap pengujian UAT (*User Acceptance Testing*) ini terdiri dari 2 faktor, yaitu : *ease of use*, *perceive of usefulness*. Kedua pengujian tersebut menitikberatkan kepada kemudahan serta kenyamanan pengguna dalam menggunakan aplikasi yang telah disiapkan. [SGH-07].

*Ease of use* adalah suatu metode pengujian yang menekankan konsep kemudahan fungsi dari suatu perangkat lunak. Jenis pengujian ini lebih condong kepada bagaimana suatu perangkat lunak dapat digunakan dengan mudah oleh pengguna. Sedangkan untuk *perceive of usefullness* merupakan suatu pengujian non fungsional yang lebih menekankan kepada apakah aplikasi atau perangkat lunak yang telah dibuat akan berguna dalam penerapannya. [SGH-07]

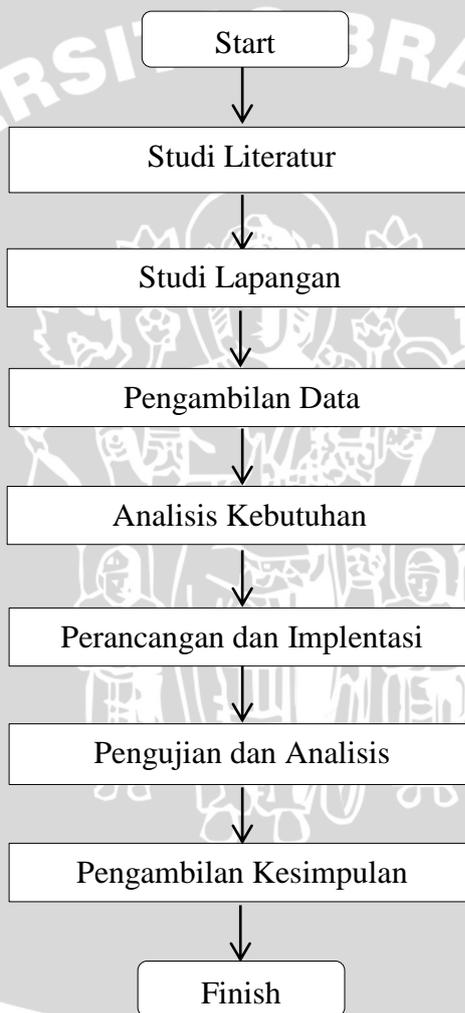
#### 2.2.4.3 Pengujian Performa

Setelah semua langkah pengujian fungsional perangkat lunak secara terstruktur dilakukan, maka perlu dilakukan pengujian sistem di lingkungan dimana dia bekerja untuk mengetahui performa dari perangkat lunak tersebut. Pengujian sistem dirancang untuk menguji waktu kinerja dari perangkat lunak dalam melakukan proses antrian transaksi. Pengujian performa melibatkan monitoring pemanfaatan sumber daya dari perangkat lunak yang diuji seperti perangkat lunak pendukung dan perangkat keras. Pengujian performa dilakukan secara spesifik sesuai dengan tipe perangkat lunak yang diuji. Pengujian performa bertujuan untuk mengungkap situasi yang menyebabkan degradasi dan kemungkinan kegagalan sistem [PRE-01].

### BAB III

## METODOLOGI PENELITIAN

Metode penelitian skripsi ini dilakukan dalam beberapa tahap, yaitu: studi literatur, studi lapangan, pengambilan data, analisa kebutuhan, perancangan sistem, implementasi, pengujian dan analisis dan yang terakhir pengambilan kesimpulan.



**Gambar 3.1 Diagram Alur Metodologi Penelitian**  
Sumber : [Perancangan]

### 3.1 Studi Literatur

#### 3.1.1 Studi Pustaka

Metode ini digunakan untuk mendapatkan dasar teori sebagai sumber acuan untuk penulisan skripsi dan pengembangan aplikasi. Teori dan pustaka yang berkaitan dengan tugas akhir ini meliputi :

1. SMS Gateway
  - AT-Command
2. Aplikasi Desktop
3. Rekayasa Perangkat Lunak
  - *Waterfall*
4. Pemograman Berorientasi Objek
  - Delphi 16 Embarcadero RAD2
5. Pengujian Perangkat Lunak
  - Teknik Pengujian
    - a. *Black-Box Testing*
    - b. Pengujian UAT
    - c. Pengujian Performa

Studi pustaka menjelaskan dasar teori yang digunakan sebagai penunjang dan pendukung penulisan skripsi. Teori penunjang dan pendukung skripsi ini meliputi *SMS Gateway*, *AT-Command*, Design dan perancangan sistem dengan pemrograman *delphi*. Sumber atau referensi yang digunakan antara lain buku, jurnal, laporan penelitian, dan bantuan mesin pencari (*search engine*) internet.

#### 3.1.2 Studi Lapangan

Pada metode ini penulis mendatangi langsung konter “First Cell” guna menanyakan masalah apa saja yang dihadapi, kebutuhan apa saja yang diinginkan, serta menanyakan produk apa saja yang dijual sebagai bahan pembuatan database sistem. Dari permasalahan yang telah didapat, dapat membangun sistem yang mana bisa menjawab segala kebutuhan serta masalah yang yang dihadapi.

### 3.2 Metode Pengambilan Data

Metode ini digunakan untuk mendapatkan data sebagai acuan untuk pengembangan pengembangan perangkat lunak. Data yang dimaksud adalah data berupa informasi mengenai harga beli, harga jual pulsa, kode transaksi, kode regional, dan segala informasi yang akan digunakan untuk implementasi database. Data-data tersebut digunakan sebagai penunjang dari *design* dan *implementasi* dari suatu perangkat lunak yang dibuat.

### 3.3 Analisis Kebutuhan

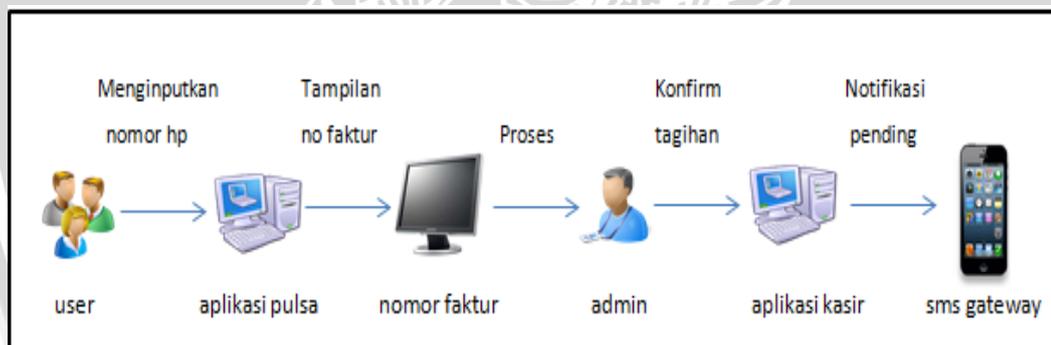
Analisis kebutuhan pada perangkat lunak ini terdiri dari kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional perangkat lunak meliputi analisis spesifikasi perangkat lunak. Metode analisis menggunakan usecase diagram untuk mendeskripsikan kebutuhan-kebutuhan dan fungsionalitas sistem dari perspektif *user*. Analisis kebutuhan dilakukan dengan mengidentifikasi semua kebutuhan (*requirements*) sistem yang kemudian akan dimodelkan dalam diagram *use case*. Kebutuhan fungsional dan non fungsional yang nantinya akan disediakan oleh aplikasi ini antara lain adalah :

1. Aplikasi pada komputer ini digunakan untuk proses transaksi pengiriman pulsa dengan fasilitas tampilan desktop yang memudahkan *user* dalam bertransaksi. Serta fitur tambah transaksi yang memungkinkan *user* dapat membeli pulsa lebih dari 1 transaksi.
2. Aplikasi ini memanfaatkan nomor faktur sebagai output transaksi yang berfungsi sebagai metode pembayaran yang menyimpan segala transaksi yang dilakukan.
3. Aplikasi ini menggunakan 2 buah PC sebagai *server* (PC utama) dan *client* (PC Kasir) yang berfungsi memproses no faktur yang ada diperoleh pada aplikasi pada saat proses transaksi.
4. Notifikasi pending menggunakan *SMS Gateway* untuk mengirimkan sms kepada *user* (pembeli) apabila transaksi yang dilakukan mengalami pending atau gagal transaksi.

### 3.4 Perancangan dan Implementasi

Perancangan arsitektur sistem adalah tahap dimana penulis mulai merancang suatu sistem yang mampu memenuhi semua kebutuhan fungsional aplikasi dalam tugas akhir ini. Teori-teori dari pustaka dan data dari sample digabungkan dengan ilmu yang didapat diimplementasikan untuk merancang serta mengembangkan suatu aplikasi *SMS Gateway* sebagai notifikasi pending pada aplikasi desktop penjualan pulsa. Perancangan sistem berdasarkan *Object Oriented Analysis* dan *Object Oriented Design* yaitu menggunakan pemodelan DFD (*Data Flow Diagram*).

Gambaran umum aplikasi ini merupakan sebuah sistem dimana segala aktifitas transaksi pengisian pulsa dilakukan langsung oleh pembeli tanpa perantara penjual. Output yang didapat pada aplikasi utama berupa nomor faktur sebagai data transaksi yang akan dikonfirmasi oleh penjual melalui aplikasi kasir. Gambaran umum perancangan sistem aplikasi desktop penjualan pulsa dengan nomor faktur sebagai antrian proses ditunjukkan pada gambar 3.2.



**Gambar 3.2 Rancangan Sistem Aplikasi Desktop Penjualan Pulsa**  
Sumber : [Perancangan]

Basis data yang dimiliki pada sistem ini hanya satu yang terhubung dengan komputer utama. Sedangkan untuk komputer kasir terhubung melalui kabel LAN dengan komputer utama. Jaringan komputer dan basis data yang digunakan akan ditunjukkan pada gambar 3.3.



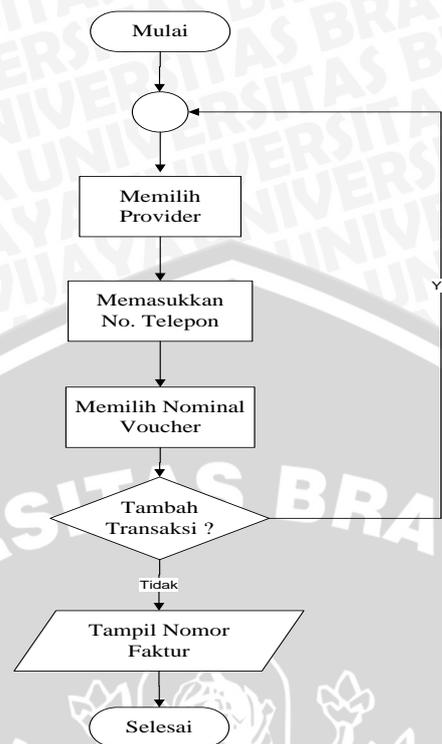
**Gambar 3.3 Diagram Alir Jaringan Komputer dan Basis Data Sistem**  
**Sumber : [Perancangan]**

Implementasi aplikasi dilakukan dengan mengacu kepada perancangan aplikasi. Implementasi perangkat lunak dilakukan dengan menggunakan Embarcadero RAD2 dengan bahasa pemrograman delphi, menggunakan implementasi *basis data* MySQL, dan MYDAC sebagai *component* yang digunakan untuk mengakses MySQL pada IDE Embarcadero RAD 2.

Tahap-tahap yang dilakukan antara lain:

- Membuat basis data dengan *Mysql* sesuai dengan desain basis data dalam perancangan aplikasi
- Mendesain aplikasi sesuai dengan perancangan pada sistem menggunakan pemrograman Delphi
- Implementasi *SMS Gateway* menggunakan *AT Command* sebagai notifikasi pending pada transaksi pengiriman pulsa.
- Melakukan pengecekan aplikasi telah sesuai dengan kebutuhan.

Diagram alir aplikasi desktop penjualan Pulsa dengan *SMS Gateway* ini akan ditunjukkan pada Gambar 3.4.



**Gambar 3.4 Diagram Alir implementasi aplikasi desktop**  
**Sumber : [Perancangan]**

Dalam proses pengembangan perangkat lunak menggunakan *waterfall model* yang pertama dilakukan adalah menganalisa kebutuhan *user*. Setelah kebutuhan *user* benar-benar dipahami maka dilanjutkan proses selanjutnya yaitu dengan merancang aplikasi desktop penjualan pulsa dan database. Proses selanjutnya dapat dilakukan apabila proses sebelumnya telah selesai. Proses selanjutnya adalah mengimplementasikan aplikasi dengan menggunakan *SMS Gateway* sebagai proses pengiriman *SMS* kepada server dan juga pengiriman *SMS* kepada *user* apabila transaksi yang dilakukan mengalami pending atau gagal transaksi. Proses terakhir adalah menguji aplikasi desktop penjualan pulsa dengan menggunakan *SMS Gateway*.

### 3.5 Pengujian dan Analisis

Pengujian dari perangkat lunak ini berkaitan dengan pengujian sistem (*black box*). Analisis sistem dilakukan dengan menguji fungsional dari aplikasi yang dibuat. Pengujian fungsional berhubungan dengan segala fitur-fitur yang ada

pada aplikasi dan berjalan sesuai dengan kebutuhan sistem. Sedangkan pengujian non fungsional menggunakan pengujian UAT (*User Acceptability Testing*) dan pengujian performa.

Pengujian UAT digunakan untuk mendapatkan konfirmasi atau kepastian melalui *trial* atau *review* dari pihak yang terkait sesuai dengan *requirement* yang telah disetujui sebelumnya. Sedangkan pengujian performa digunakan untuk mengetahui berapa lama waktu tunggu dari setiap transaksi yang dilakukan dibandingkan dengan sistem lama atau menggunakan transaksi secara manual. Dari pengujian ini diharapkan dapat diketahui apakah sistem yang dibangun dapat melakukan semua proses yang ada sesuai dengan yang diharapkan *user*.

### 3.6 Pengambilan Kesimpulan

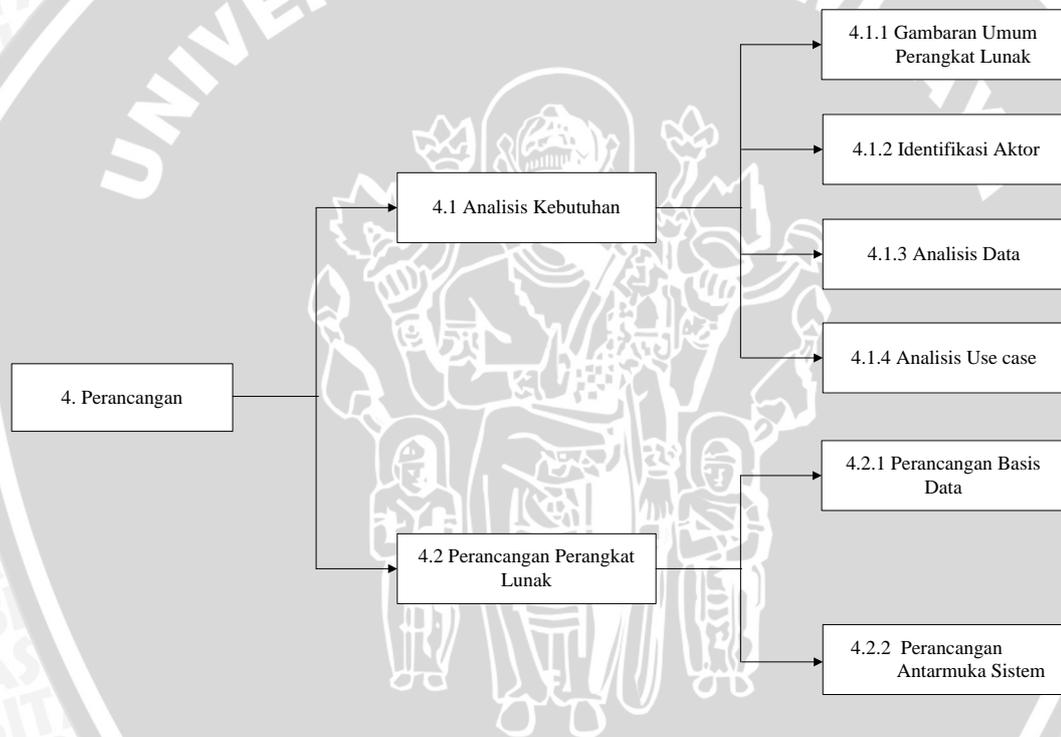
Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi dan pengujian sistem telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan aplikasi selanjutnya.



## BAB IV PERANCANGAN DAN IMPLEMENTASI

Bab ini membahas mengenai perancangan aplikasi desktop dengan nomor faktur sebagai antrian proses pada aplikasi penjualan pulsa. Perancangan yang dilakukan meliputi dua tahap. Proses analisis kebutuhan dilakukan pada tahap pertama dan proses perancangan perangkat lunak dilakukan pada tahap kedua.

Tahap-tahap perancangan yang dilakukan pada aplikasi desktop dengan nomor faktur sebagai antrian proses pada aplikasi penjualan pulsa dijelaskan pada Gambar 4.1 berikut ini.



**Gambar 4.1 Diagram Blok Perancangan**  
**Sumber : [Perancangan]**

### 4.1. Analisis Kebutuhan

Proses analisis kebutuhan ini mengacu pada hasil pengumpulan data, pemahaman dan penetapan kebutuhan-kebutuhan yang ingin didapatkan oleh pengguna. Proses analisis kebutuhan ini diawali dengan penjabaran gambaran umum aplikasi desktop, identifikasi aktor yang terlibat, analisis data yang akan



disimpan, penjabaran tentang daftar kebutuhan, sistem *SMS Gateway* yang berjalan, dan kemudian memodelkannya ke dalam diagram *use case*. Analisis kebutuhan ini bertujuan untuk menggambarkan kebutuhan – kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna.

#### 4.1.1 Gambaran Umum Aplikasi Desktop Penjualan Pulsa

Perangkat lunak pada sistem ini memiliki 3 fungsi utama yaitu, aplikasi utama untuk pengisian pulsa langsung oleh pembeli, aplikasi kasir sebagai administrator pemrosesan transaksi, serta aplikasi *SMS Gateway* sebagai mediasi koneksi ke server. Aplikasi pulsa pada sistem ini bertindak sebagai *downline* pada sebuah server pulsa.

##### a. Aplikasi Pulsa

Aplikasi *user* dari perangkat lunak ini adalah aplikasi utama berupa form transaksi yang ditujukan langsung kepada *user* (pembeli) dalam melakukan pengisian pulsa. Fungsi utama yang dapat dilakukan oleh aplikasi ini adalah :

1. Pilihan operator  
Operasi ini bertujuan untuk melakukan pilihan operator terhadap transaksi yang akan dilakukan.
2. Penginputan nomor transaksi  
Operasi ini bertujuan untuk melakukan penginputan nomor handphone sesuai dengan nomor tujuan dari transaksi yang dilakukan.
3. Pilihan Voucher  
Operasi ini bertujuan untuk melakukan penginputan jumlah nominal pulsa yang akan dikirim kepada nomor handphone sesuai dengan nomor tujuan dari transaksi yang dilakukan.
4. Tambah transaksi  
Operasi ini bertujuan untuk melakukan proses penambahan transaksi dalam satu nomor faktur dan satu kali transaksi.
5. Hapus transaksi  
Operasi ini bertujuan untuk melakukan penghapusan terhadap transaksi yang dilakukan.

#### 6. No Faktur

Operasi ini bertujuan untuk mendapatkan nomor faktur dari transaksi yang dilakukan. Nomor faktur tersebut berfungsi sebagai nomor antrian dari proses transaksi yang dilakukan.

#### b. Aplikasi Administrator

Aplikasi administrator (kasir) dari perangkat lunak ini adalah aplikasi pemrosesan transaksi yang telah dilakukan dengan antrian sesuai nomor faktur yang tertera pada setiap transaksi. Aplikasi ini juga bersifat sebagai sistem antrian terhadap transaksi yang dilakukan oleh *user* (pembeli) dan belum terkonfirmasi pembayarannya. Proses-proses utama yang terdapat dalam aplikasi administrator (kasir) adalah sebagai berikut :

##### 1. Login

Operasi ini bertujuan untuk melakukan proses masuk pada aplikasi kasir yang hanya dapat dilakukan oleh admin dengan memasukkan *username* dan *password* yang telah ditentukan.

##### 2. Antrian Transaksi

Operasi ini bertujuan untuk melakukan penginputan nomor faktur dari transaksi yang dilakukan oleh *user*.

##### 3. Data Transaksi

Operasi ini bertujuan untuk mengecek data transaksi yang masuk atau melalui tahap antrian. Pada form aplikasi ini berisi tentang nomor faktur, kode voucher, no handphone, serta waktu transaksi sesuai dengan pilihan status dari transaksi.

##### 4. Setting

Operasi ini bertujuan sebagai form setting untuk mensetting semua kebutuhan aplikasi perangkat lunak.

### c. Aplikasi SMS Gateway

Aplikasi *SMS Gateway* pada aplikasi desktop penjualan pulsa ini merupakan aplikasi yang dapat melakukan proses pengiriman dan penerimaan sms dari server. Proses-proses utama yang terdapat dalam aplikasi *SMS Gateway* adalah sebagai berikut:

#### 1. Connect / Disconnect

Fitur ini berfungsi sebagai tombol on/of untuk mengkoneksikan aplikasi dengan modem.

#### 2. Pilihan Port

Fitur ini berfungsi sebagai aplikasi pembaca port-port yang tersambung pada modem.

#### 3. Memo 1 (SMS Masuk)

Fitur ini berfungsi menampilkan *SMS* masuk dan terkirim sesuai dengan format *SMS* pada database.

#### 4. Memo 2 (SMS Keluar)

Fitur ini berfungsi menampilkan proses AT-Command yang berjalan pada aplikasi *SMS Gateway*.

### 4.1.2 Identifikasi Aktor

Tahap ini adalah tahap untuk melakukan identifikasi terhadap aktor yang akan berinteraksi dengan aplikasi desktop dengan menggunakan *SMS gateway* sebagai notifikasi pending. Tabel 4.1 memperlihatkan aktor – aktor yang terlibat beserta penjelasannya masing-masing yang merupakan hasil dari proses identifikasi aktor.

**Tabel 4.1 Identifikasi Aktor**

Aktor	Deskripsi
<i>User</i>	<i>User</i> (pembeli) adalah pengguna yang dapat menggunakan semua fitur yang ada pada aplikasi utama, diantaranya melakukan pemilihan operator, penginputan nomor handphone, pemilihan nominal pulsa, tambah transaksi, serta

	hapus transaksi.
<b>Administrator</b>	Administrator adalah <i>user</i> yang bertindak sebagai kasir pada aplikasi administrator. Fungsi kasir menginputkan nomor faktur yang didapat <i>user</i> (pembeli) dari aplikasi utama dan memproses antrian transaksi. Pada tahap ini, segala transaksi yang telah dilakukan dan tercatat pada nomor faktur dikonfirmasi dan dilakukan proses pentransferan pulsa sesuai dengan data pada nomor faktur.

Sumber : [Perancangan]

#### 4.1.3 Analisis Data

Analisis data bertujuan untuk mendapatkan struktur penyimpanan data yang dibutuhkan perangkat lunak. Struktur penyimpanan data pada perangkat lunak ini disusun berdasarkan analisis data sebagai berikut :

- Data transaksi yang terdiri dari id, pilihan operator, nomor pengguna, nominal pulsa, dan jumlah transaksi. Data ini berfungsi untuk menyimpan data terhadap proses atau transaksi yang dilakukan.
- Data faktur yang terdiri dari id, jumlah transaksi, dan jumlah nominal pembayaran. Data ini berfungsi untuk menyimpan data transaksi kedalam satu nomor faktur dalam satu kali proses.
- Data Antrian yang terdiri dari nomor faktur, nomor transaksi, jumlah transaksi, dan jumlah nominal pembayaran dalam 1 nomor faktur. Data ini berfungsi untuk menyimpan data antrian transaksi untuk diproses dan dikonfirmasi pada aplikasi kasir.
- Data pemrosesan transaksi yang terdiri dari fungsi kirim dan menerima SMS dari dan menuju server. Data ini diproses pada aplikasi SMS gateway berdasarkan format transaksi pada data transaksi.

#### 4.1.4 Analisis Use case

Analisis use case merupakan gambaran dari analisis kebutuhan dari suatu sistem. Pada sistem aplikasi ini dibagi mejadi 2 bagian, yaitu daftar kebutuhan fungsional dan daftar kebutuhan non fungsional. Pada daftar kebutuhan fungsional akan dispesifikasikan menjadi tiga, yaitu spesifikasi kebutuhan fungsional *user* (pembeli), spesifikasi kebutuhan fungsional administrator dan spesifikasi kebutuhan fungsional *SMS Gateway*. Spesifikasi kebutuhan fungsional *user* ditunjukkan pada Tabel 4.2 dibawah ini.

**Tabel 4.2 Spesifikasi kebutuhan fungsional *user* (pembeli)**

Nomor SRS	Kebutuhan	Use Case
SRS_001_01	Perangkat lunak harus mampu menampilkan halaman utama aplikasi dengan tombol isi pulsa sebagai proses utama	Halaman utama
SRS_001_02	Perangkat lunak harus mampu memilih operator sesuai dengan transaksi yang diinputkan <i>user</i> .	Pilihan Operator
SRS_001_03	Perangkat lunak harus mampu memproses inputan nomor hanphone sesuai dengan transaksi yang diinputkan <i>user</i>	Penginputan Nomor
SRS_001_04	Perangkat lunak harus mampu memilih nominal vocher dari operator yang dipilih sesuai inputan dari <i>user</i> .	Pilihan Voucher
SRS_001_05	Perangkat lunak harus mampu melakukan tambah transaksi dari beberapa pilihan operator	Tambah Transaksi
SRS_001_06	Perangkat lunak harus mampu melakukan hapus / batal	Hapus Transaksi

	transaksi dari jumlah transaksi	
SRS_001_07	Perangkat lunak harus mampu mencetak nomor faktur dari setiap transaksi dari <i>user</i>	Tampilkan Nomor Faktur

Sumber : [Perancangan]

Spesifikasi kebutuhan fungsional kasir ditunjukkan pada Tabel 4.3.

**Tabel 4.3 Spesifikasi kebutuhan fungsional kasir**

Nomor SRS	Kebutuhan	Use Case
SRS_002_01	Perangkat lunak harus mampu mengakses database dan memungkinkan administrator melakukan login dengan <i>username</i> dan <i>password</i> yang telah dilakukan	Login
SRS_002_02	Perangkat lunak harus menyediakan fasilitas untuk penginputan nomor faktur dari transaksi yang dilakukan oleh <i>user</i> pada aplikasi utama	Antrian Transaksi
SRS_002_03	Perangkat lunak harus menyediakan fasilitas form pengecekan data transaksi yang masuk atau melalui tahap antrian	Data Transaksi
SRS_002_04	Perangkat lunak harus menyediakan fasilitas sistem setting yang <i>handle</i> segala kebutuhan sistem yang ada pada aplikasi utama	Setting

Sumber : [Perancangan]

Spesifikasi kebutuhan fungsional *SMS Gateway* ditunjukkan pada Tabel 4.4.

**Tabel 4.4 Spesifikasi kebutuhan fungsional kasir**

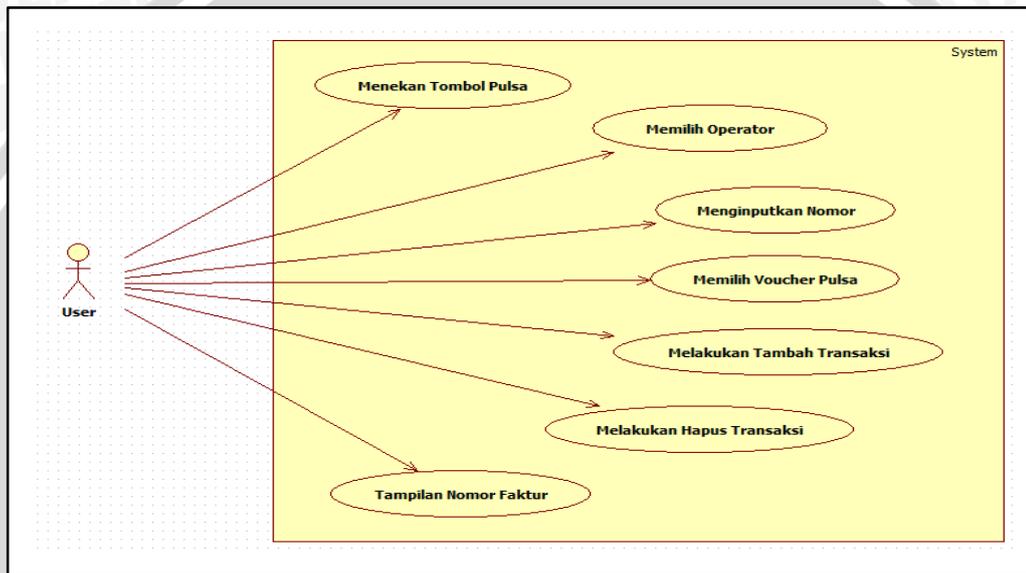
Nomor SRS	Kebutuhan	Use Case
SRS_003_01	Perangkat lunak harus mampu menyambungkan aplikasi dengan modem sehingga port pada aplikasi dapat membaca port-port pada modem	Connect
SRS_003_02	Perangkat lunak harus mampu berjalan dan mengirimkan <i>SMS</i> sesuai dengan format transaksi pada server dan menyimpan data pada database	Kirim <i>SMS</i>
SRS_003_03	Perangkat lunak harus mampu berjalan dan menerima <i>SMS</i> dari server dengan format yang tersedia dan dapat menyimpan proses data pada database	Menerima <i>SMS</i>

**Sumber : [Perancangan]**

Diagram *use case* adalah salah satu diagram untuk memodelkan aspek perilaku sistem. Diagram *use case* menunjukkan sekumpulan *use case*, aktor, dan hubungannya. *Use case* merupakan fungsionalitas dari sistem yang diinisialisasi oleh aktor. Pemodelan diagram *use case* yang menggambarkan fungsionalitas perangkat lunak *SMS Gateway* pada aplikasi desktop penjualan pulsa dibagi menjadi tiga, yaitu diagram *use case* untuk aplikasi *user* (pembeli), diagram *use case* untuk aplikasi *administrator* (kasir) dan diagram *use case* untuk aplikasi *SMS gateway*.

#### 4.1.4.1 Diagram Use Case Aplikasi Pulsa

Diagram *use case* ini melibatkan pembeli sebagai aktor utama dan empat buah *use case*. Empat buah *use case* ini termasuk dalam bagian *software SMS Gateway* pada aplikasi desktop penjualan pulsa. Empat buah *use case* ini juga akan disertai dengan skenario *use case* untuk menjelaskan rangkaian aktifitas yang terjadi di *use case* tersebut. Diagram *use case* untuk aplikasi pasien ditunjukkan pada Gambar 4.2 berikut ini .



**Gambar 4.2 Diagram Use Case User (pembeli)**  
**Sumber : [Perancangan]**

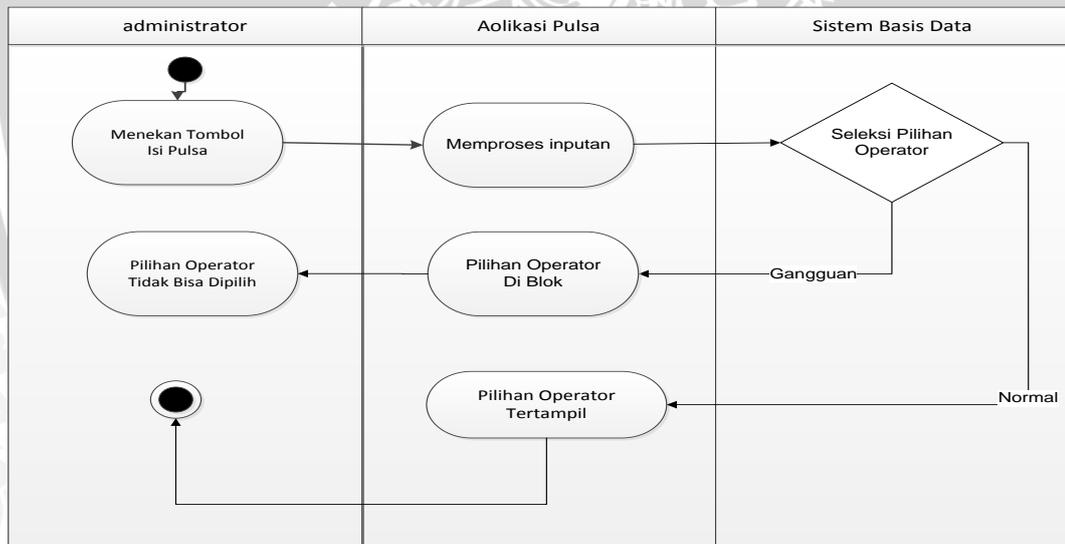
#### 1. Skenario Use Case Menekan Tombol Isi Pulsa

**Tabel 4.5 Use case menekan tombol isi pulsa**

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_01
Nama	Menekan Tombol Isi Pulsa
Tujuan	Memulai proses transaksi pulsa
Deskripsi	Use case ini menjelaskan bagaimana user pertama kali masuk pada halaman utama dan memulai transaksi

	dengan menekan tombol isi pulsa sebagai awal proses transaksi
Aktor	User
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Aplikasi Destop</i> penjualan pulsa telah berjalan dan terkoneksi dengan database.
Aksi Aktor	Reaksi Sistem
User menekan tombol isi pulsa	Sistem menerima inputan dan mengecek pilihan operator yang tersedia berdasarkan data pada database.
Kondisi Akhir	Sistem menampilkan halaman pilihan operator

Sumber : [Perancangan]



Gambar 4.3 Diagram Aktifitas Halaman Utama

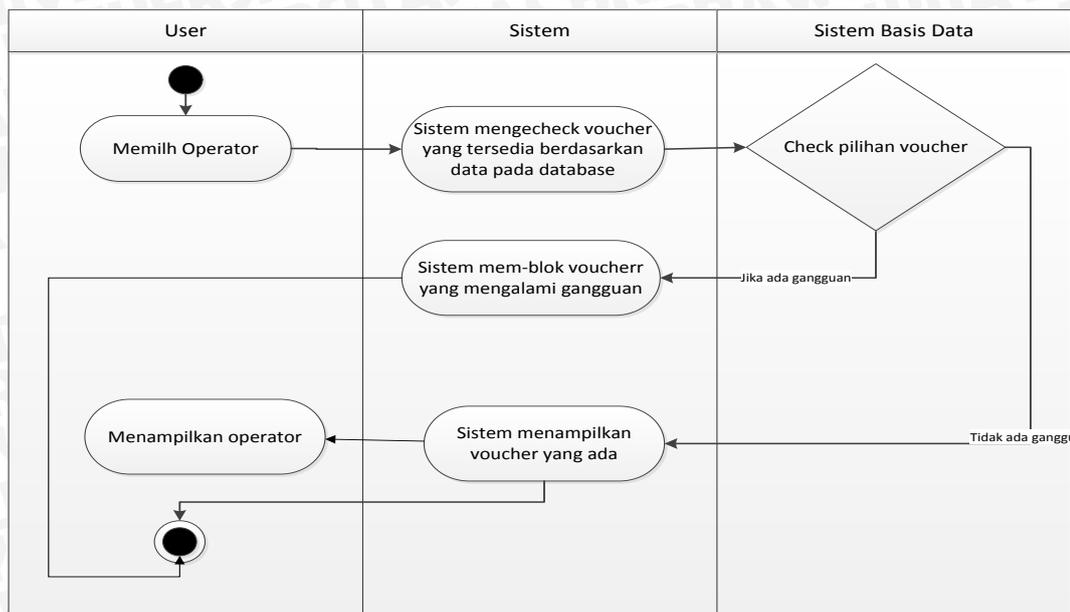
Sumber : [Perancangan]

## 2. Skenario *Use Case* Memilih Operator

**Tabel 4.6** *Use case* Melakukan Pemilihan Operator

Skenario Kasus Pada Sistem	
Nomor <i>Use Case</i>	SRS_001_02
Nama	Melakukan Pemilihan Operator
Tujuan	Untuk memilih operator yang diinginkan
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana <i>user</i> melakukan pemilihan operator sesuai dengan transaksi yang diinginkan
Aktor	<i>User</i>
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Aplikasi Destop</i> penjualan pulsa telah berjalan dan terkoneksi dengan database.
Aksi Aktor	Reaksi Sistem
<i>User</i> memilih operator pada form pilihan operator	Sistem menerima inputan dan mengecek pilihan voucher yang tersedia berdasarkan data pada database.
Kondisi Akhir	Sistem menampilkan operator terpilih dengan nominal voucher yang tersedia

Sumber : [Perancangan]



**Gambar 4.4 Diagram Aktifitas Memilih Operator**  
**Sumber : [Perancangan]**

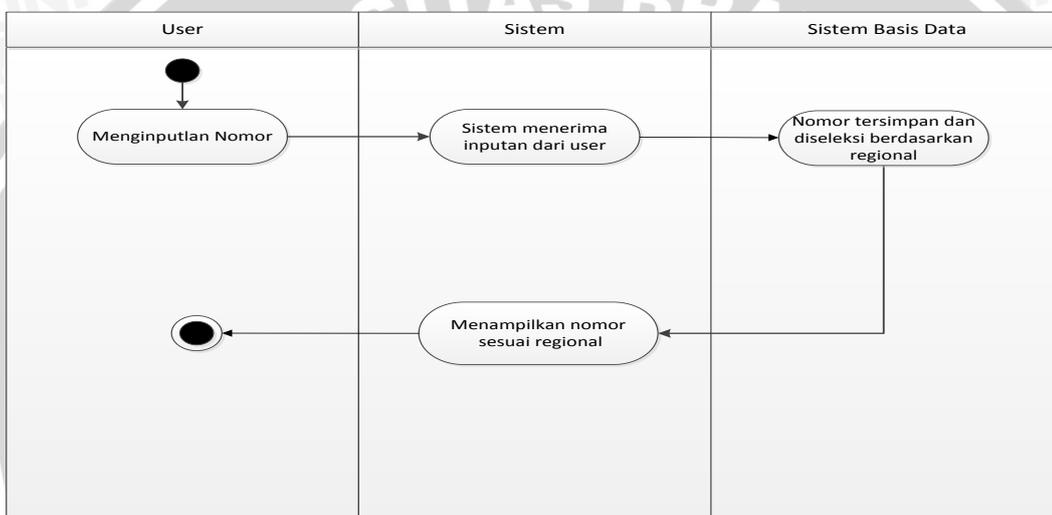
**3. Skenario Use Case Menginputkan Nomor**

**Tabel 4.7 Use case Melakukan Penginputan Nomor**

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_03
Nama	Melakukan Penginputan Nomor Tujuan
Tujuan	Untuk memasukkan nomor transaksi
Deskripsi	Use Case ini menjelaskan bagaimana user melakukan penginputan nomor tujuan untuk proses pengisian pulsa sesuai dengan transaksi yang diinginkan
Aktor	User (pembeli)
Skenario Utama	
Kondisi Awal	Perangkat lunak Aplikasi Destop penjualan pulsa telah berjalan dan terkoneksi dengan database.

Aksi Aktor	Reaksi Sistem
User menginputkan nomor handphone	Sistem menerima inputan yang dilakukan oleh user. Pada operator telkomsel terjadi penyeleksian nomor (msisdn) sesuai dengan kode regional.
Kondisi Akhir	Sistem menampilkan nomor handphone yang akan dipakai untuk melakukan transaksi pulsa

Sumber : [Perancangan]



Gambar 4.5 Diagram Aktifitas Menginputkan Nomor

Sumber : [Perancangan]

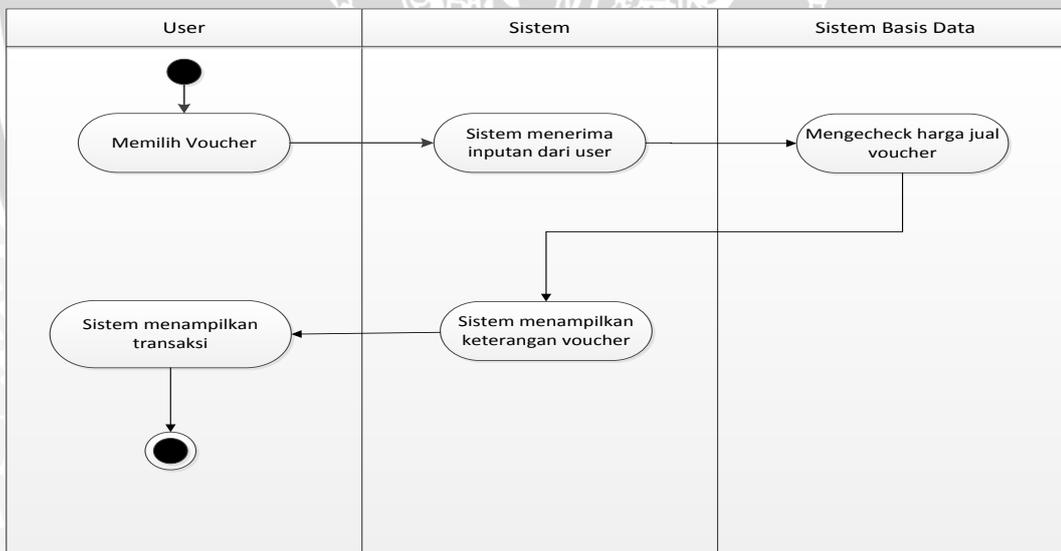
#### 4. Skenario Use Case Memilih Voucher Pulsa

Tabel 4.8 Use case Melakukan Pemilihan Nominal Voucher

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_04
Nama	Melakukan Pilihan Nominal Voucher
Tujuan	Untuk melakukan pemilihan nominal voucher
Deskripsi	Use Case ini menjelaskan bagaimana user melakukan

	pemilihan voucher untuk proses pengisian pulsa
Aktor	User
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Aplikasi Destop</i> penjualan pulsa telah berjalan dan terkoneksi dengan database.
Aksi Aktor	Reaksi Sistem
User memilih nominal voucher	Sistem menerima inputan yang dilakukan user. Database sistem menyeleksi kode voucher yang tersedia sesuai dengan data pada database sistem
Kondisi Akhir	Sistem menampilkan harga dan nominal voucher yang telah dipilih

Sumber : [Perancangan]



Gambar 4.6 Diagram Aktifitas Memilih Voucher Pulsa

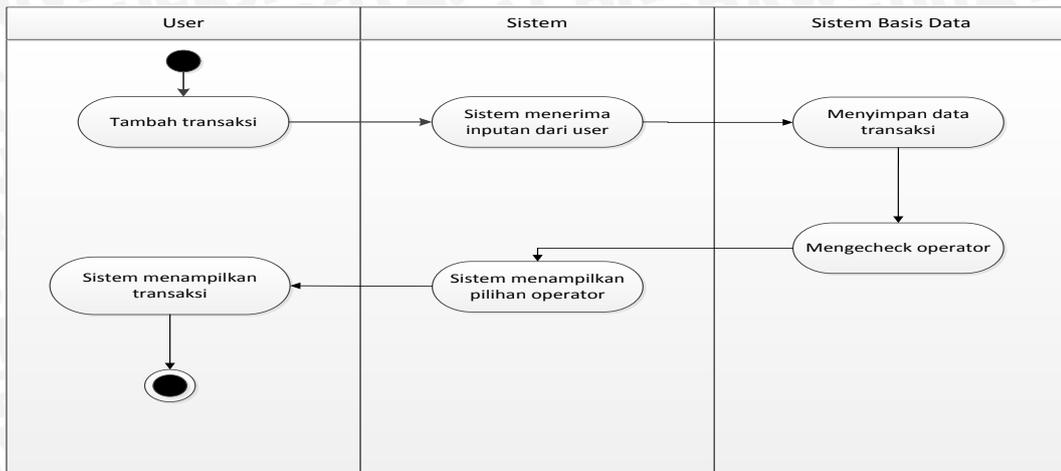
Sumber : [Perancangan]

## 5. Skenario *Use Case* Tambah Transaksi

**Tabel 4.9** *Use case* Melakukan Tambah Transaksi

Skenario Kasus Pada Sistem	
Nomor <i>Use Case</i>	SRS_001_05
Nama	Tambah Transaksi
Tujuan	Untuk melakukan tambah transaksi
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana <i>user</i> melakukan tambah transaksi pada proses pengisian pulsa
Aktor	<i>User</i> (pembeli)
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Aplikasi Desktop</i> penjualan pulsa telah berjalan dan terkoneksi dengan database.
Aksi Aktor	Reaksi Sistem
<i>User</i> menekan tombol tambah pada halaman operator yang tampil pada desktop	Sistem menerima inputan yang dilakukan <i>user</i> dan memproses inputan dengan menampilkan kembali halaman pilihan operator dan menyimpan transaksi sebelumnya dalam satu nomor faktur
Kondisi Akhir	Sistem kembali menampilkan halaman operator

**Sumber :** [Perancangan]



**Gambar 4.7 Diagram Aktifitas Tambah Transaksi**  
 Sumber : [Perancangan]

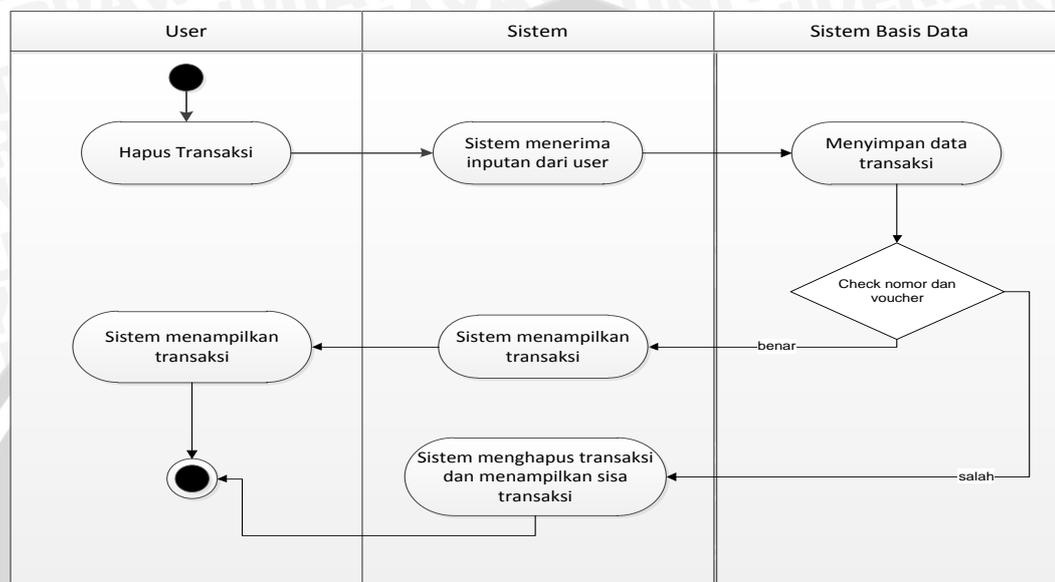
**6. Skenario Use Case Hapus Transaksi**

**Tabel 4.10 Use case Hapus Transaksi**

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_06
Nama	Hapus Transaksi
Tujuan	Untuk melakukan hapus / batal transaksi
Deskripsi	Use Case ini menjelaskan bagaimana user melakukan hapus atau batal transaksi pada proses pengisian pulsa
Aktor	User (pembeli)
Skenario Utama	
Kondisi Awal	Perangkat lunak Aplikasi Destop penjualan pulsa telah berjalan dan terkoneksi dengan database.
Aksi Aktor	Reaksi Sistem
User menekan tombol hapus	Sistem menerima inputan yang dilakukan user dan memproses inputan dengan menghapus transaksi

	sebelumnya dilakukan
Kondisi Akhir	Sistem menghapus transaksi yang dilakukan <i>user</i>

Sumber : [Perancangan]



Gambar 4.8 Diagram Aktifitas Hapus Transaksi  
Sumber : [Perancangan]

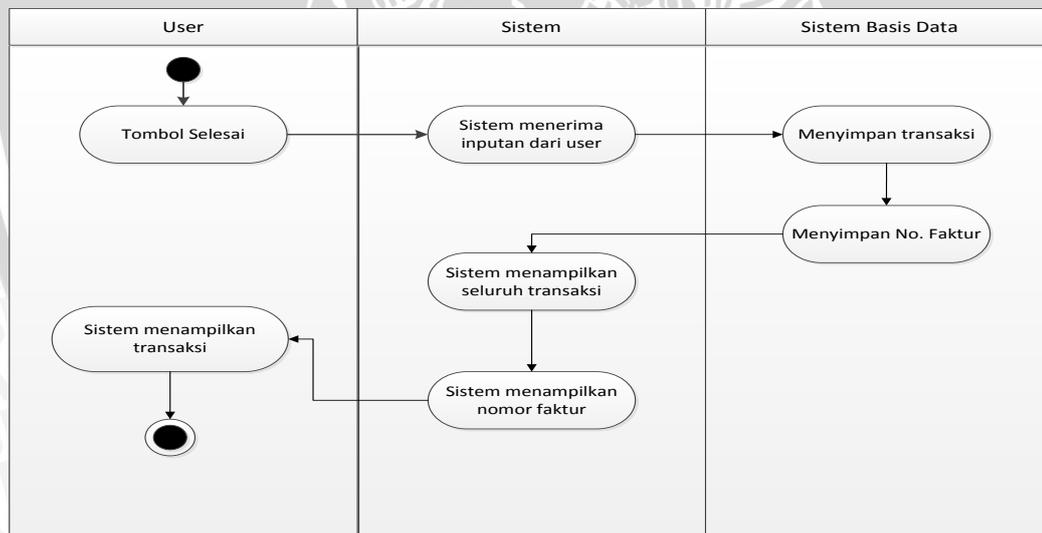
### 7. Skenario Use Case Menerima No. Faktur

Tabel 4.11 Use case Menerima Nomor Faktur

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_07
Nama	Menerima Nomor Faktur
Tujuan	Sebagai nomor antrian untuk diproses
Deskripsi	Use Case ini menjelaskan bagaimana <i>user</i> mendapatkan nomor faktur dari transaksi yang dilakukan. Nomor faktur tersebut digunakan sebagai nomor antrian untuk diproses pada aplikasi kasir
Aktor	<i>User</i> (pembeli)

Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Aplikasi Destop</i> penjualan pulsa telah berjalan dan terkoneksi dengan database. Sistem mencetak nomor faktur dari transaksi yang dilakukan oleh <i>user</i>
Aksi Aktor	Reaksi Sistem
<i>User</i> menekan tombol selesai pada form halaman konfirmasi	Sistem menerima inputan yang dilakukan <i>user</i> dan memproses inputan dengan menampilkan kembali halaman pilihan operator dan menyimpan transaksi sebelumnya dalam satu nomor faktur
Kondisi Akhir	Sistem kembali menampilkan halaman operator

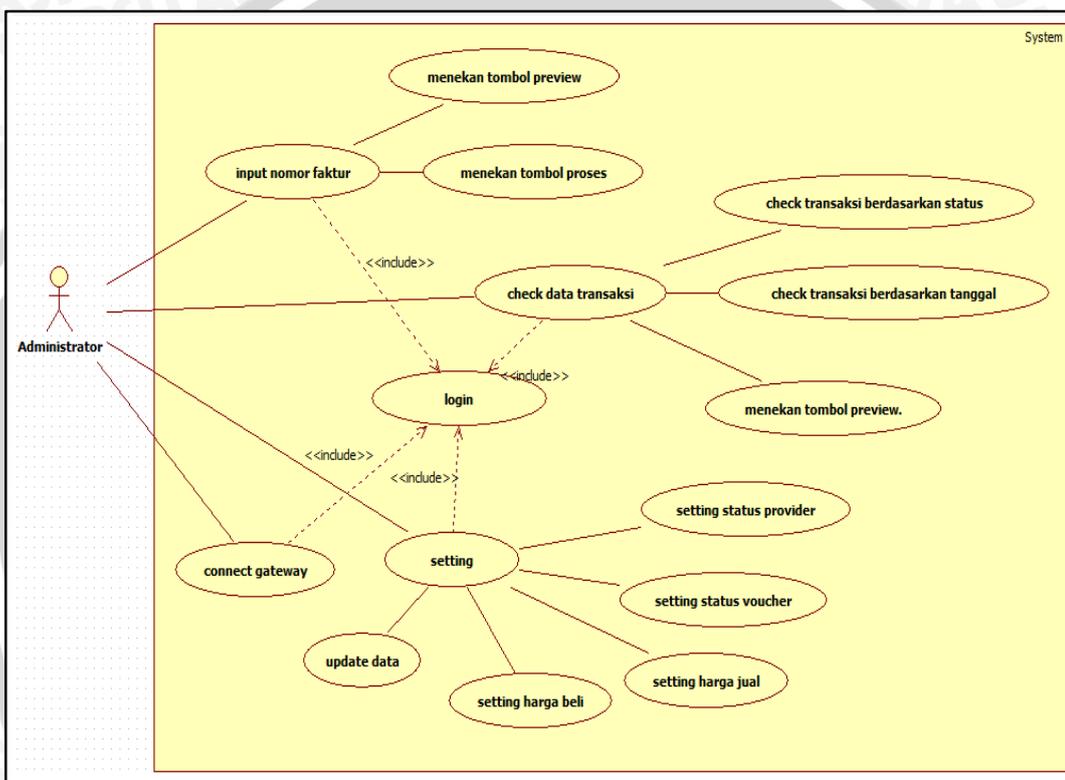
Sumber : [Perancangan]



Gambar 4.9 Diagram Aktifitas Tampilan No. Faktur  
 Sumber : [Perancangan]

#### 4.1.4.2 Diagram Use Case Aplikasi Kasir

Diagram *use case* ini melibatkan administrator (kasir) sebagai aktor dan empat buah *use case* didalamnya. Empat buah *use case* ini termasuk dalam bagian *software SMS Gateway* pada aplikasi desktop penjualan pulsa. Empat buah *use case* ini juga disertai dengan skenario *use case* untuk menjelaskan rangkaian aktifitas yang terjadi pada *use case*. Penjelasan *diagram use case* ditunjukkan pada Gambar 4.10 berikut ini:



Gambar 4. 10 Diagram Use Case Administrator (kasir)  
Sumber : [Perancangan]

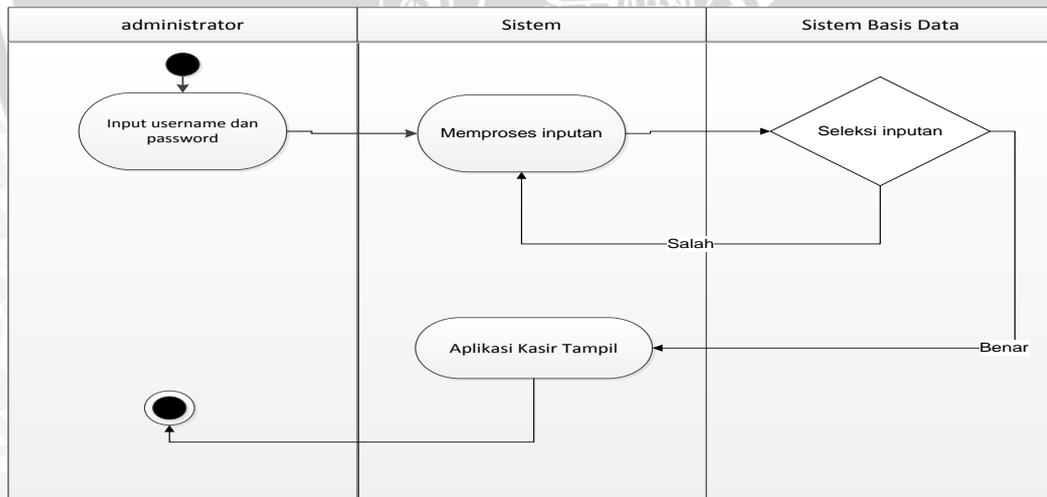
#### 1. Skenario Use Case Login

Tabel 4.12 Use case Login

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_01
Nama	Login
Tujuan	Masuk halaman aplikasi kasir

Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana proses administrator menginputkan <i>username</i> dan password untuk masuk pada halaman aplikasi kasir
Aktor	administrator
Skenario Utama	
Kondisi Awal	Perangkat lunak aplikasi utama telah berjalan dan terkoneksi dengan database
Aksi Aktor	Reaksi Sistem
<i>Administrator</i> input id <i>username</i> dan password	Sistem menerima inputan dan menyeleksi id <i>username</i> dan password berdasarkan data pada database.
Kondisi Akhir	Sistem memproses inputan dan menampilkan halaman aplikasi kasir.

Sumber : [Perancangan]



Gambar 4.11 Diagram Aktifitas Login

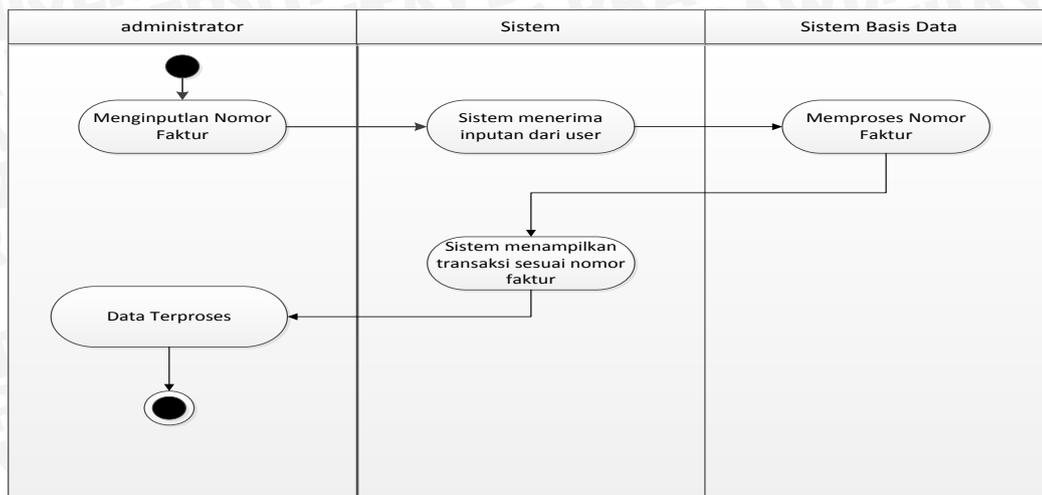
Sumber : [Perancangan]

## 2. Skenario *Use Case* Menginputkan Nomor Faktur

**Tabel 4.13** *Use case* Menginputkan Nomor Faktur

Skenario Kasus Pada Sistem	
Nomor <i>Use Case</i>	SRS_002_02
Nama	Menginputkan Nomor Faktur
Tujuan	Menampilkan dan memproses transaksi
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana proses administrator menginputkan nomor faktur dan memproses transaksi
Aktor	Administator (kasir)
Skenario Utama	
Kondisi Awal	Aplikasi kasir telah berjalan dan terkoneksi dengan database.
Aksi Aktor	Reaksi Sistem
Administrator (kasir) menginputkan nomor faktur, apabila telah tampil, kasir menekan tombol proses	Sistem menyeleksi jumlah transaksi berdasarkan inputan nomor faktur. Transaksi yang muncul hanya transaksi yang tercatat pada nomor faktur yang telah diproses.
Kondisi Akhir	<ol style="list-style-type: none"> <li>1. Sistem memproses nomor faktur dan menampilkan status transaksi.</li> <li>2. Aplikasi kasir memproses transaksi dengan mengirimkan format transaksi melalui aplikasi gateway.</li> </ol>

**Sumber :** [Perancangan]



**Gambar 4.12 Diagram Aktifitas Menginputkan Nomor Faktur**  
 Sumber : [Perancangan]

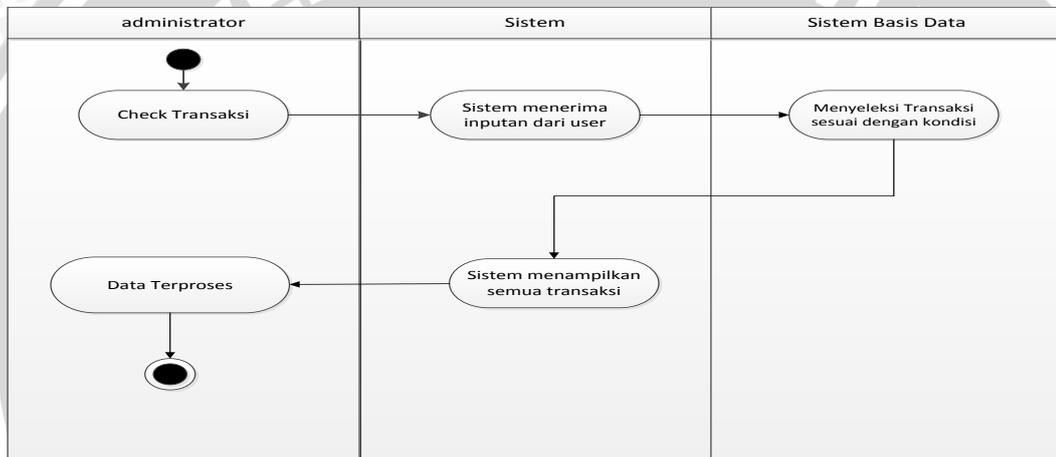
**3. Skenario Use Case Check Data Transaksi**

**Tabel 4.14 Use case Check Data Transaksi**

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_03
Nama	Check Data Transaksi
Tujuan	Mengecheck status transaksi yang dilakukan
Deskripsi	Use Case ini menjelaskan bagaimana administrator mengecek status transaksi yang dilakukan.
Aktor	Administator (kasir)
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>adminitrator kasir</i> telah berjalan dan terkoneksi dengan database
Aksi Aktor	Reaksi Sistem

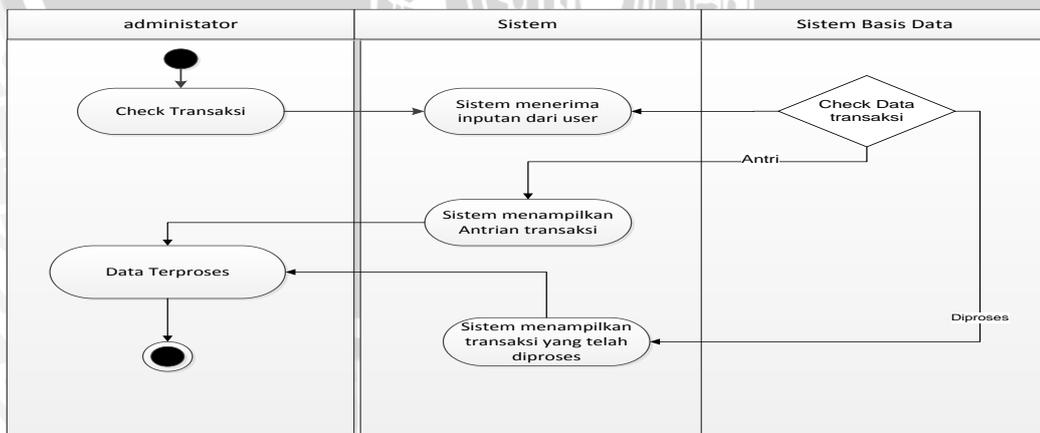
<p><i>Administrator</i> (kasir) mengecek data transaksi berdasarkan status transaksi</p>	<p>Sistem menyeleksi data transaksi berdasarkan inputan yang dilakukan administrator berdasarkan tanggal ataupun status transaksi</p>
<p>Kondisi Akhir</p>	<p>Sistem menampilkan status transaksi yang telah dilakukan. Baik yang sudah berhasil ataupun masih pending berdasarkan no faktur, kode voucher, nomor handphone, dan waktu transaksi</p>

Sumber : [Perancangan]



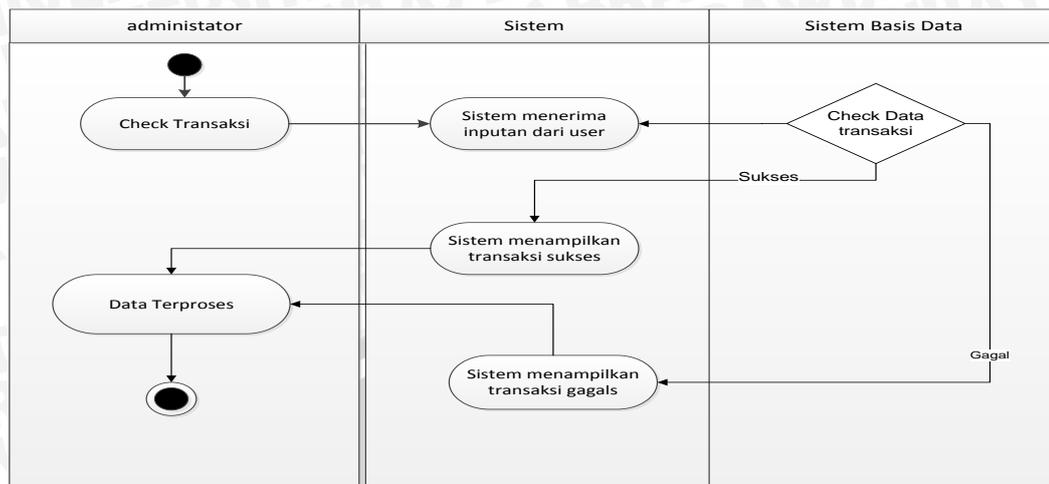
Gambar 4.13 Diagram Aktifitas Check (Semua) Data Transaksi

Sumber : [Perancangan]



Gambar 4.14 Diagram Aktifitas Check Data Transaksi Antrian dan Diproses

Sumber : [Perancangan]



**Gambar 4.15 Diagram Aktifitas Check Data Transaksi Sukses dan Gagal**  
**Sumber : [Perancangan]**

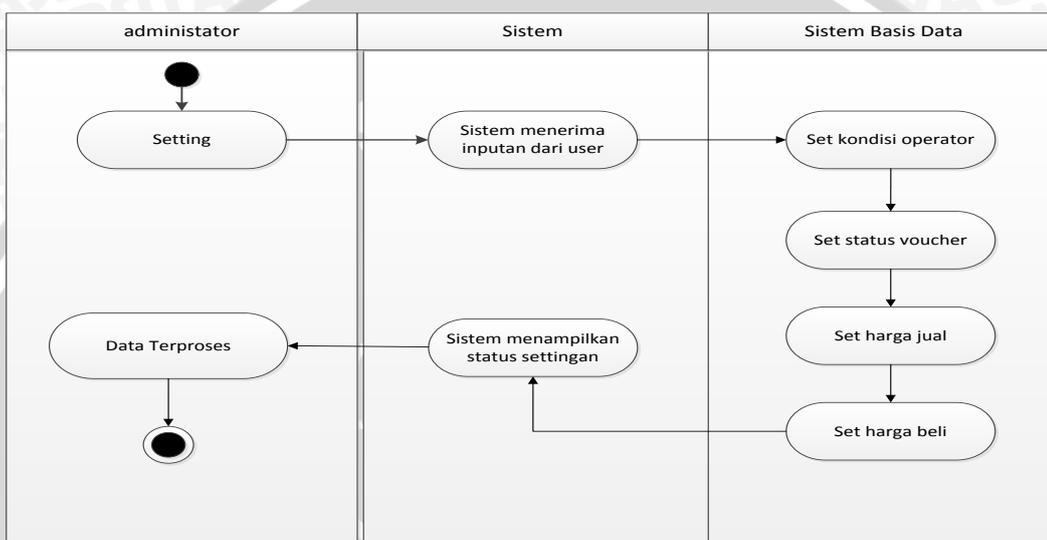
#### 4. Skenario Use Case Setting Transaksi

**Tabel 4.15 Use case Setting Transaksi**

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_04
Nama	Setting Transaksi
Tujuan	Menyeting segala kebutuhan sistem
Deskripsi	Use Case ini menjelaskan bagaimana administrator menyeting segala kebutuhan penjualan aplikasi pulsa baik dari ketersediaan voucher sampai dengan harga
Aktor	Administator (kasir)
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>adminitrator kasir</i> telah berjalan dan terkoneksi dengan database
Aksi Aktor	Reaksi Sistem
<i>Administrator</i> (kasir) menyeting kebutuhan	Sistem memproses settingan inputan administrator dan menyimpannya pada database sistem sesuai dengan

sistem	tabel dan kolom yang dibutuhkan
Kondisi Akhir	Sistem menyetting status provider dan voucher yang tersedia, serta mengatur harga jual dan beli voucher

Sumber : [Perancangan]



Gambar 4.16 Diagram Aktifitas Setting  
Sumber : [Perancangan]

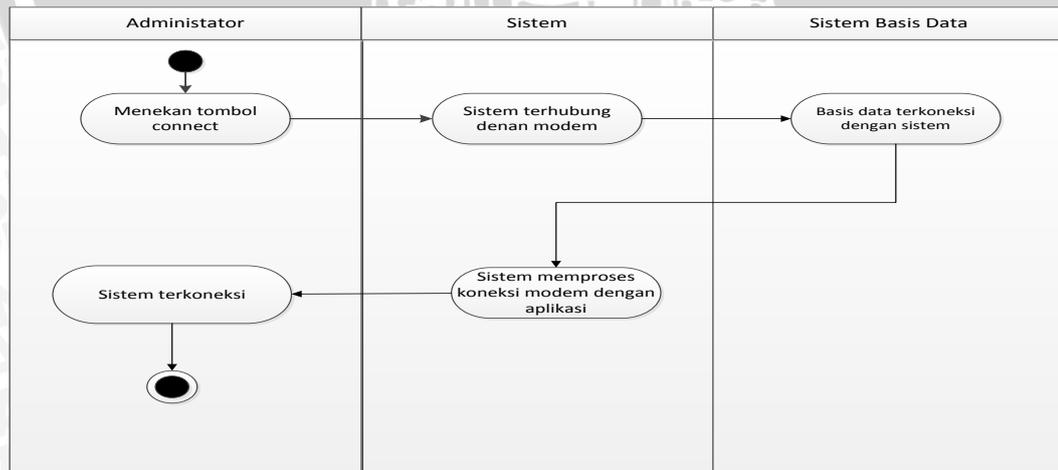
### 5. Skenario Use Case Connect Gateway

Tabel 4.16 Use case Connect Gateway

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_003_01
Nama	Connect
Tujuan	Mengkoneksikan aplikasi dengan modem
Deskripsi	Use Case ini menjelaskan bagaimana aplikasi terkoneksi dengan modem dan membaca port-port yang ada pada modem.

Aktor	Administator (kasir)
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>SMS gateway</i> terkoneksi dengan database dan modem sebagai media kirim dan menerima <i>SMS</i>
Aksi Aktor	Reaksi Sistem
<i>Administrator</i> (kasir) menekan tombol connect	Aplikasi terkoneksi dengan database dan tersambung dengan modem sebagai media pengiriman dan penerimaan <i>SMS</i> . Aplikasi berhasil membaca port-port pada modem.
Kondisi Akhir	<ol style="list-style-type: none"> <li>1. Aplikasi terkoneksi dengan sistem database dan modem.</li> <li>2. Aplikasi gateway dapat mengirmkan <i>SMS</i> ke server dengan format transaksi dari aplikasi kasir</li> <li>3. Aplikasi gateway dapat menerima <i>SMS</i> dari server dan menerukan pada pengguna.</li> </ol>

Sumber : [Perancangan]



Gambar 4.17 Diagram Aktifitas Connect  
 Sumber : [Perancangan]

Pada daftar kebutuhan non fungsional sistem perangkat lunak untuk aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses terbagi menjadi 3. Daftar kebutuhan non fungsional ditinjau dari segi *availability*, *usablity*, dan *security*. Selengkapnya dijelaskan pada tabel 4.17 berikut ini.

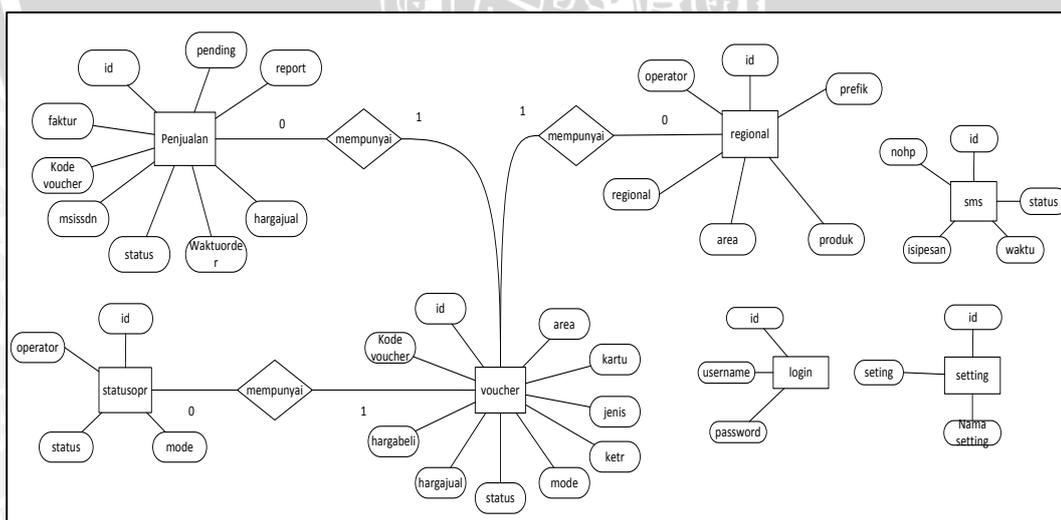
**Tabel 4.17 Spesifikasi Kebutuhan Non Fungsional**

Parameter	Deskripsi Kebutuhan
Availability	Perangkat lunak harus dapat beroperasi selama jam kerja
Usability	Perangkat lunak harus dapat digunakan dengan mudah oleh pengguna
Security	Perangkat lunak harus dapat menyimpan data dengan valid

## 4.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak memiliki tiga tahap, yaitu perancangan basis data, perancangan algoritma, dan perancangan antarmuka pengguna dari perangkat lunak aplikasi desktop dengan SMS gateway sebagai notifikasi pending.

### 4.2.1 Perancangan Basis Data



**Gambar 4.18 Perancangan Basis Data**  
Sumber : [Perancangan]

Gambar 4.18 menunjukkan atribut dari tabel-tabel yang ada pada database sistem perangkat lunak. Berikut ini merupakan struktur tabel serta keterangan masing masing tabel dan *field* yang ada pada *database*. Tabel aplikasi penjualan pulsa ini memiliki 4 tabel yaitu tabel penjualan, status operator, regional, voucher, dan tabel penjualan. Struktur tabel penjualan ditunjukkan pada Tabel 4.18.

**Tabel 4.18 Struktur tabel penjualan**

No.	Nama Field	Tipe	Lebar	Keterangan
1	Id	int	11	Id transaksi
2	faktur	int	11	Nomor antrian
4	kodevoucher	varchar	10	Kode voucher pulsa
5	msisdn	varchar	25	Nomor awalan ( <i>regional</i> )
6	status	datetime		Status voucher
7	waktuorder	tinyint	4	Waktu melakukan transaksi
8	hargajual	Int	11	Harga jual voucher
9	report	Int	11	Status report SMS
10	pending	int	11	Status pending

Entitas *status operator* merepresentasikan setting dari operator yang berisi setting dari id, operator, status, dan mode. Struktur tabel *status operator* ditunjukkan pada Tabel 4.19.

**Tabel 4.19 Struktur tabel status operator**

No.	Nama Field	Tipe	Lebar	Keterangan
1	Id	tinyint	4	Id transaksi
2	operator	varchar	20	Jenis operator
4	status	tinyint	4	Status operator
5	mode	tinyint	4	Status voucher

Entitas *regional* merepresentasikan kode wilayah dari operator yang mencakup id, operator, regional, area, produk dan prefix. Struktur tabel *regional* ditunjukkan pada Tabel 4.20.

Tabel 4.20 Struktur tabel *regional*

No.	Nama Field	Type	Lebar	Keterangan
1	Id	integer	11	Id transaksi
2	operator	varchar	20	Jenis operator
3	regional	varchar	20	Nomor regional operator
4	area	varchar	20	Wilayah kota
5	produk	varchar	20	Nama operator
6	prefik	varchar	20	Nomor awalan operator

Entitas *voucher* merepresentasikan seluruh keterangan terhadap voucher yang mencakup id, kodevoucher, harga beli, harga jual, status, mode, keterangan, jenis, kartu, dan area dari voucher tersebut. Struktur tabel *regional* ditunjukkan pada Tabel 4.21.

Tabel 4.21 Struktur tabel *voucher*

No.	Nama Field	Type	Lebar	Keterangan												
	Id	integer	11	Id transaksi												
	kodevoucher	varchar	10	Kode voucher pulsa												
	hargabeli	integer	11	Harga beli voucher												
	hargajual	integer	11	Harga jual voucher												
	status	tinyint	3	Status operator												
	mode	tinyint	3	Status voucher												
	ke <tr> <td></td> <td>jenis</td> <td>varchar</td> <td>20</td> <td>Jenis voucher</td> </tr> <tr> <td></td> <td>kartu</td> <td>varchar</td> <td>20</td> <td>Nama kartu</td> </tr> <tr> <td></td> <td>area</td> <td>varchar</td> <td>20</td> <td>Area kartu</td> </tr>		jenis	varchar	20	Jenis voucher		kartu	varchar	20	Nama kartu		area	varchar	20	Area kartu
	jenis	varchar	20	Jenis voucher												
	kartu	varchar	20	Nama kartu												
	area	varchar	20	Area kartu												

Entitas SMS mempresentasikan seluruh keterangan terhadap tabel sms yang mencakup id, nohp, isipesan, waktu, status, statustrx, mode dari tabel sms tersebut. Struktur tabel sms ditunjukkan pada Tabel 4.22.

Tabel 4.22 Struktur Tabel Sms

No.	Nama Field	Tipe	Lebar	Keterangan
	Id	int	11	Id transaksi
	nohp	varchar	255	No hp transaksi
	isipesan	varchar	255	Format transaksi
	waktu	datetime		Waktu order
	status	tinyint	11	Status order
	statustrx	tinyint	4	Status transaksi
	mode	Varchar	10	Inbox / Outbox

Entitas *setting* mempresentasikan seluruh keterangan terhadap tabel setting yang mencakup id, setting, namasetting dari tabel setting tersebut. Struktur tabel *setting* ditunjukkan pada Tabel 4.23.

Tabel 4.23 Struktur Tabel Setting

No.	Nama Field	Tipe	Lebar	Keterangan
	Id	Int	11	Id transaksi
	setting	Varchar	255	Nomor Center
	namasetting	Varchar	20	Nama setting

Entitas *login* mempresentasikan seluruh keterangan terhadap tabel login yang mencakup id, *username*, *password* dari tabel login tersebut. Struktur tabel *login* ditunjukkan pada Tabel 4.24.

Tabel 4.24 Struktur Tabel Login

No.	Nama Field	Tipe	Lebar	Keterangan
	id	Int	11	id administrator
	username	Varchar	255	username administrator
	password	Varchar	20	password administrator

#### 4.2.2 Perancangan Antarmuka sistem

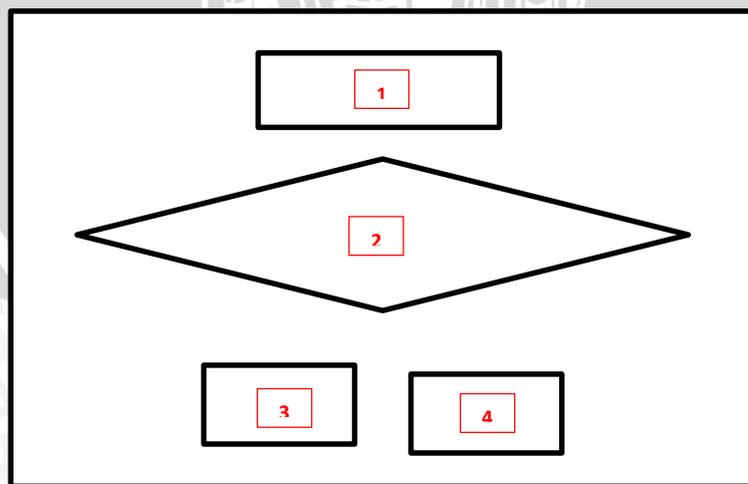
Pada bagian ini dijelaskan tentang perancangan antarmuka perangkat lunak aplikasi desktop menggunakan *SMS gateway* sebagai notifikasi pending. Antarmuka perangkat lunak ini akan digunakan oleh pengguna untuk berinteraksi dengan sistem secara langsung. Antarmuka perangkat lunak ini dibagi menjadi tiga bagian, yaitu antarmuka aplikasi utama untuk interfase yang digunakan *user* (pembeli), aplikasi *administrator* sebagai aplikasi kasir dan aplikasi *SMS Gateway* sebagai pemrosesan sms masuk dan keluar.

##### 4.2.2.1 Rancangan Antarmuka Aplikasi Pulsa

Pada Perancangan ini terdiri dari beberapa tampilan halaman utama. Diantaranya adalah antarmuka tampilan utama, antarmuka pilihan operator, antarmuka menginputkan nomor tujuan dan pilihan voucher, antarmuka cetak nomor faktur.

#### 1. Antarmuka Tampilan Utama

Antarmuka tampilan utama untuk aplikasi *user* berupa sebuah halaman *desktop*. Antarmuka aplikasi ini terdiri dari logo konter, animasi, serta tombol isi pulsa sebagai proses awal untuk melakukan transaksi pulsa. Antarmuka tampilan utama mengacu pada spesifikasi kebutuhan pada SRS\_001\_1 ditunjukkan pada gambar 4.19 berikut ini.



Gambar 4.19 Tampilan Antarmuka Utama

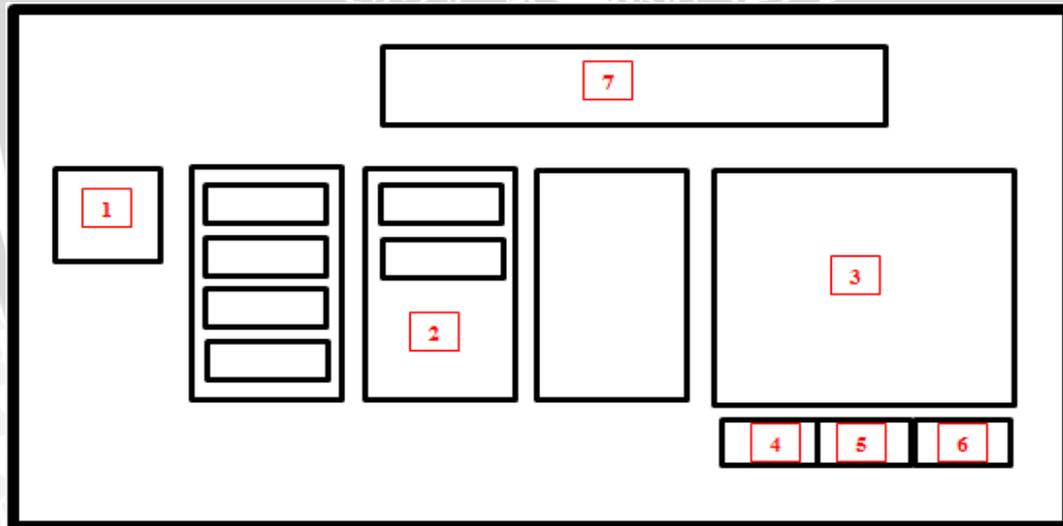
Sumber: [Perancangan]

Gambar 4.19 mempunyai keterangan sebagai berikut :

- Gambar 1 : image logo counter yang bersangkutan
- Gambar 2 : animasi berjalan pada saat aplikasi aktif
- Gambar 3 : tombol isi pulsa sebagai fungsi untuk memulai transaksi
- Gambar 4 : tombol admin untuk proses login menuju aplikasi kasir

## 2. Antarmuka Tampilan Menginputkan Nomor Tujuan dan Voucher

Halaman tampilan voucher merupakan salah satu antarmuka pengguna untuk menginputkan nomor tujuan dan memilih voucher dari operator terpilih. Halaman ini berfungsi untuk mempermudah pengguna dalam menentukan voucher apa saja yang akan diambil selama transaksi. Perancangan aplikasi ini mengacu pada spesifikasi kebutuhan SRS\_001\_03 dan SRS\_001\_04. Gambar-4.20 akan menunjukkan perancangan tampilan antarmuka dari halaman tampilan operator aplikasi desktop penjualan pulsa.



**Gambar 4.20 Tampilan Antarmuka XL**

**Sumber: [Perancangan]**

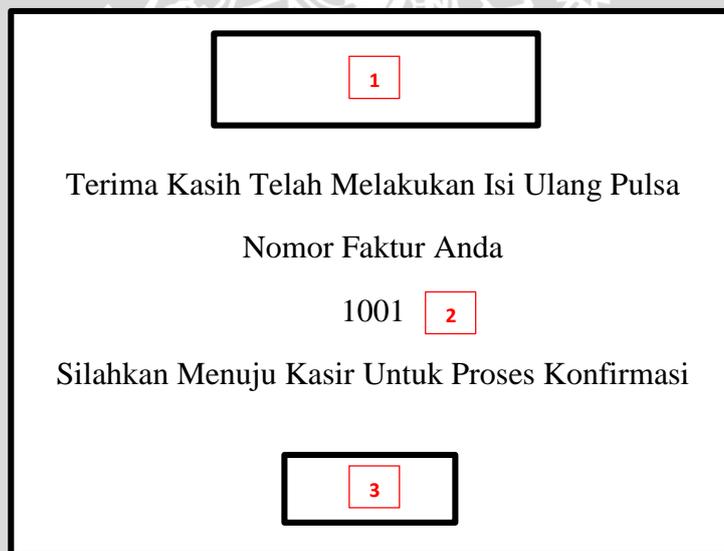
Gambar 4.20 mempunyai keterangan sebagai berikut :

- Gambar 1 : image operator yang terpilih
- Gambar 2 : image voucher dengan keterangan jenis voucher (reguler)
- Gambar 3 : image voucher dengan keterangan jenis voucher (SMS)

- Gambar 4 : image voucher dengan keterangan jenis voucher (internet)
- Gambar 5 : label keterangan yang menampilkan keterangan dari voucher yang dipilih dan harga jual yang dikenakan
- Gambar 6 : tombol kembali untuk menampilkan halaman sebelumnya
- Gambar 7 : tombol tambah transaksi yang menampilkan kembali halaman pilihan operator
- Gambar 8 : tombol selesai yang menampilkan keseluruhan transaksi
- Gambar 9 : field untuk menginputkan nomor tujuan.

### 3. Antarmuka Tampilan Nomor Faktur

Tampilan nomor faktur adalah antarmuka pengguna untuk melihat dari transaksi yang telah dilakukan. Perancangan aplikasi ini mengacu pada spesifikasi kebutuhan SRS\_001\_7. Gambar 4.21 akan menunjukkan perancangan tampilan nomor faktur pada aplikasi desktop penjualan pulsa.



**Gambar 4.21 Tampilan Antarmuka Nomor Faktur**

**Sumber: [Perancangan]**

Gambar 4.21 mempunyai keterangan sebagai berikut :

- Gambar 1 : image logo counter yang bersangkutan
- Gambar 2 : label edit menampilkan nomor faktur dari transaksi yang dilakukan

Gambar 3 : tombol selesai untuk mengakhiri transaksi yang dilakukan dan kembali ke halaman utama

#### 4.2.2.2 Rancangan Antarmuka Aplikasi Kasir

Pada Perancangan ini terdiri dari beberapa tampilan halaman. Diantaranya adalah antarmuka tampilan antrian transaksi, tampilan data transaksi, dan tampilan antarmuka setting. Ketiga rancangan antarmuka tersebut mengacu pada kebutuhan sistem perangkat lunak yang akan ditunjukkan pada gambar 4.22-4.24 berikut ini.

##### 1. Rancangan Antarmuka Login

The image shows a login form within a window. At the top right of the window are three control buttons: minimize, maximize, and close (labeled 'x'). Below these are four callout boxes with numbers 1 through 4:

- 1: A small square icon in the top right corner of the form area.
- 2: A text input field labeled "Username :".
- 3: A text input field labeled "Password :".
- 4: A button labeled "Login".

**Gambar 4.22 Tampilan Antarmuka Login**

**Sumber: [Perancangan]**

Gambar 4.22 mempunyai keterangan sebagai berikut :

Gambar 1 : Image logo counter yang bersangkutan

Gambar 2 : Field username untuk mengisi nama pengguna

Gambar 3 : Field Password untuk mengisi kata sandi

Gambar 4 : Tombol login untuk menjalankan proses masuk ke aplikasi kasir

## 2. Rancangan Antarmuka Antrian Transaksi

The image shows a user interface for a transaction queue. At the top, there are three input fields labeled 1, 2, and 3. Below these is a table with four columns, labeled 4. At the bottom, there is a label 'Total Pembelian Rp. -,' followed by a small box labeled 5.

**Gambar 4.23 Tampilan Antarmuka Antrian Transaksi**

**Sumber: [Perancangan]**

Gambar 4.23 mempunyai keterangan sebagai berikut :

- Gambar 1 : Field input nomor faktur untuk mengecek transaksi sesuai nomor faktur
- Gambar 2 : Tombol preview untuk menampilkan transaksi berdasarkan nomor faktur
- Gambar 3 : Tombol proses untuk memproses transaksi dari nomor faktur yang telah di inputkan
- Gambar 4 : Kolom transaksi untuk menampilkan transaksi terdiri dari no faktur, kode voucher, msisdn, dan harga jual
- Gambar 5 : Label edit jumlah tagihan dari total transaksi yang dilakukan sesuai dengan nomor faktur

### 3. Rancangan Antarmuka SMS Gateway

The screenshot displays the SMS Gateway configuration interface. It features several input fields and buttons:

- Port:** A dropdown menu (labeled 1) and a text input field (labeled 4).
- SMS center:** A text input field (labeled 2).
- Delay:** A text input field (labeled 3).
- Connect:** A button (labeled 6) located below the input fields.
- Message 1:** A large text area (labeled 5) for displaying outgoing messages.
- Message 2:** A large text area (labeled 6) for displaying incoming messages.

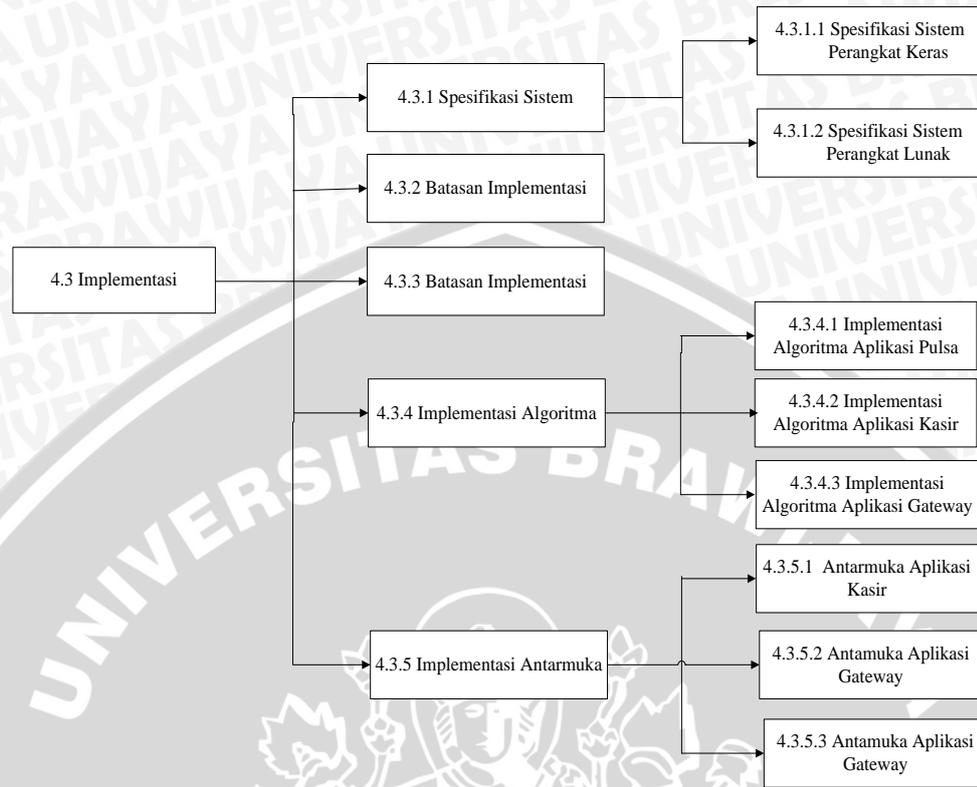
Gambar 4.24 Tampilan Antarmuka SMS Gateway

Sumber: [Perancangan]

Gambar 4.24 mempunyai keterangan sebagai berikut :

- Gambar 1 : Field comport untuk memilih port modem yang terdeteksi
- Gambar 2 : Field SMS center untuk memilih nomor center server
- Gambar 3 : Field delay untuk mensetting delay transaksi sebagai
- Gambar 4 : Tombol connect untuk mengkoneksikan modem dengan aplikasi
- Gambar 5 : Field memo 1 untuk menampilkan isi pesan terkirim
- Gambar 6 : Field memo 2 untuk menampilkan isi pesan diterima

### 4.3 Implementasi



Gambar 4.25 Diagram Blok Implementasi

Sumber : [Implementasi]

#### 4.3.1 Spesifikasi Sistem

Aplikasi desktop sistem penjualan pulsa menggunakan nomor faktur sebagai antrian proses dikembangkan dalam lingkungan implementasi yang terdiri dari perangkat keras dan perangkat lunak.

##### 4.3.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang dipakai dalam proses pengembangan Aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses dijelaskan pada Tabel 4.25.

**Tabel 4.25 Spesifikasi perangkat keras komputer**

<b>Macbook White Unibody Late 2010 (MC516ZA/A)</b>	
Processor	Intel Core 2 Duo (T8300) 2,24 GHZ
Memory (RAM)	2 GB
Harddisk	250GB (5400 RPM) Serial ATA (3 Gb/s)
Motherboard	Intel Pentium
Monitor	13.3" Widescreen 1280x800
<b>Modem Wismo Wavecom Fastrack</b>	
Model	M1306B
Jenis	GSM/GPRS : 900/1800/1900MHZ

**Sumber : [Implementasi]**

#### 4.3.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat keras yang dipakai dalam proses pengembangan Aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses dijelaskan pada Tabel 4.26.

**Tabel 4.26 Spesifikasi perangkat lunak komputer**

<b>Macbook White Unibody</b>	
Operating System	Microsoft Windows 7 32-bit
Programming Language	Delphi
Database Management System	MySQL 4.1
Integrated Development Environment	Embarcadero RAD Studio XE2

Software Development Tools	SQLyog Enterprise 8.18
----------------------------	------------------------

Sumber : [Implementasi]

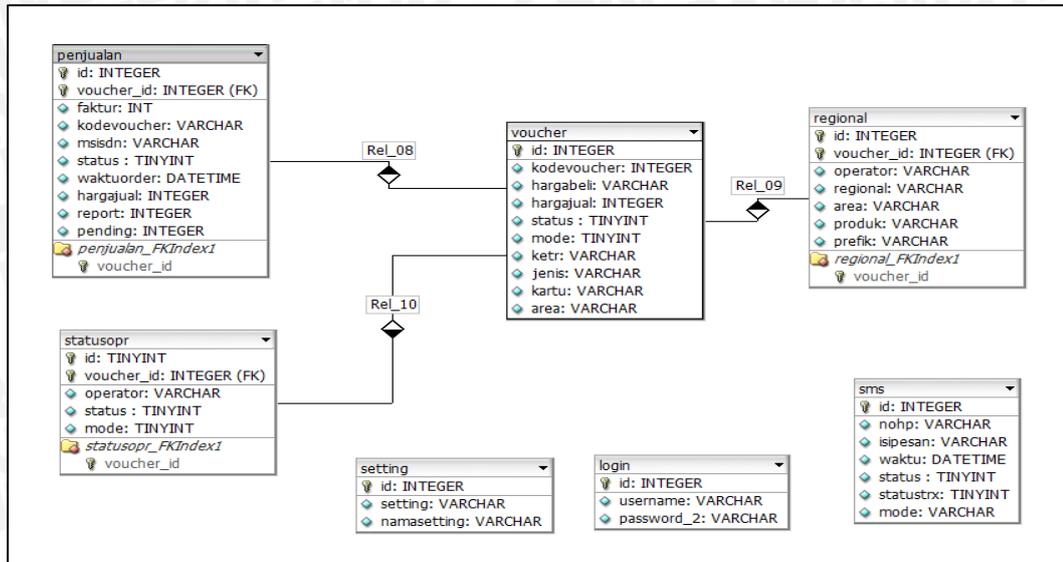
#### 4.3.2 Batasan Implementasi

Beberapa batasan dalam mengimplementasikan perangkat lunak Aplikasi Desktop Penjualan Pulsa dengan nomor faktur sebagai antrian proses sebagai berikut :

1. Perangkat lunak Aplikasi Desktop menggunakan nomor faktur sebagai antrian proses dirancang dan dijalankan dengan menggunakan *Delphi Desktop Application*.
2. Perangkat lunak Aplikasi Desktop menggunakan nomor faktur sebagai antrian proses berjalan sebagai *downline* pada sebuah server pusat.
3. *Database Management System* yang digunakan adalah MySQL.
4. Perangkat lunak *Aplikasi Desktop* menggunakan *SMS Gateway* sebagai notifikasi pending menggunakan tarif *SMS* untuk proses konfirmasi terhadap *server* dan pengiriman *SMS* kepada pembeli.

#### 4.3.3 Implementasi Basis Data

Implementasi penyimpanan data dilakukan dengan *database management system* MySQL. Hasil implementasi penyimpanan data ini berupa *script – script* SQL. Hasil implementasi SQL pada *database* ini dimodelkan dalam diagram konseptual *entity relationship*. Gambar 4.26 menggambarkan diagram konseptual *entity relationship* dari perangkat lunak



**Gambar 4.26 Diagram ER Konseptual Aplikasi Desktop Penjualan Pulsa**  
Sumber : [Implementasi]

#### 4.3.4 Implementasi Algoritma

Pada implementasi algoritma dijelaskan algoritma-algoritma pada setiap aplikasi. Algoritma-algoritma tersebut terbagi menjadi beberapa bagian. Diantaranya pada algoritma aplikasi pulsa, aplikasi kasir, dan aplikasi SMS Gateway.

##### 4.3.4.1 Algoritma Aplikasi Pulsa

Aplikasi penjualan pulsa menggunakan nomor faktur sebagai antrian proses memiliki beberapa proses (*method*) utama. Algoritma proses yang ada pada aplikasi pulsa antara lain adalah proses tampilan home, isi pulsa, pilihan operator, input nomor telepon, pilihan nominal voucher, tambah transaksi, proses selesai, proses menampilkan nomor faktur, serta proses transaksi. Implementasi algoritma isi pulsa akan direpresentasikan dalam bentuk *code* dengan bahasa pemrograman Delphi pada gambar 4.27.

##### a. Implementasi Algoritma Home (Isi Pulsa)

1.	<code>procedure THome.Button1Click(Sender: TObject);</code>
2.	<code>begin</code>
3.	<code>Floatanimation1.Enabled := False;</code>
4.	<code>pilihanoperator.qstatusopr.close;</code>
5.	<code>pilihanoperator.qstatusopr.SQL.clear;</code>

```
6.      pilihanoperator.qstatusopr.sql.add('select status from
7.      statusopr where operator="XL"');
8.      pilihanoperator.qstatusopr.Open;
9.      if
10.     pilihanoperator.qstatusopr.FieldByName('STATUS').AsString=
11.     '0' then
12.     begin
13.         pilihanoperator.xl.Enabled := False;
14.         pilihanoperator.label_xl.Visible := True;
15.     end;
16.     if
17.     pilihanoperator.qstatusopr.FieldByName('STATUS').AsString=
18.     '1' then
19.     begin
20.         pilihanoperator.xl.Enabled := True;
21.         pilihanoperator.label_xl.Visible := False;
22.     end;
23.     end;
```

**Gambar 4.27 Implementasi Algoritma Isi Pulsa**  
**Sumber : [Implementasi]**

Penjelasan implementasi algoritma *isi pulsa* dalam Gambar 4.31 yaitu:

1. Baris 6-18 merupakan proses status pilihan operator pada saat pertama kali *user* menekan tombol isi pulsa
2. Baris 9-23 merupakan algoritma operator indosat dimana apabila inputan string pada database menunjukkan angka 0 maka pilihan operator indosat tidak akan muncul, akan terganti dengan label indosat yang menandakan stok kosong atau gangguan. Apabila inputan string pada database menunjukkan angka 1 maka pilihan operator indosat akan muncul, dan label indosat yang menandakan stok kosong atau gangguan tidak muncul.

#### 4.3.4.2 Algoritma Aplikasi Kasir

Algoritma proses yang ada pada aplikasi kasir antara lain adalah proses login, *antrian proses*, proses *antrian transaksi*, proses *setting*. Implementasi algoritma login akan direpresentasikan dalam bentuk *code* dengan bahasa pemrograman Delphi pada gambar 4.28 dan 4.29.

##### a. Implementasi Algoritma Halaman Login

```
1. procedure TForm2.LOGINClick(Sender: TObject);
2. begin
3.   if (edit1.Text <> '') and (edit2.Text <> '') then
4.     begin
5.       qlogin.Close;
6.       qlogin.SQL.Clear;
7.       qlogin.SQL.Add('select count(*) as n from login where
8. username="'
9.   + string(Edit1.Text) + '" and password="' +
10. string(Edit2.Text) + '"');
11.      qlogin.Open;
12.      if qlogin.FieldByName('n').AsString <> '0' then
13.        begin
14.          form2.Hide;
15.          form1.Show;
16.          home.Hide;
17.          formfaktur.hide;
18.        end;
19.      if qlogin.FieldByName('n').AsString = '0' then
20.        begin
21.          ShowMessage('Maaf Username dan Password Anda
22. Salah');
23.          Edit1.Clear;
24.          Edit2.Clear;
25.        end;
26.      end;
27.    end;
```

**Gambar 4.28 Implementasi Algoritma Halaman Login**  
Sumber : [Implementasi]

Penjelasan implementasi algoritma halaman login dalam gambar 4.28 yaitu:

1. Baris 3-16 menjelaskan proses login administrator pada saat masuk pada aplikasi kasir. Administrator memasukkan *username* dan password sesuai dengan id yang didapat. Apabila berhasil login, administrator akan masuk pada form 1 yaitu aplikasi kasir.
2. Baris 17-24 menjelaskan proses dimana administrator salah memasukkan *username* dan password sesuai dengan id yang didapat. Pada proses ini akan menampilkan notifikasi bahwa *username* dan password yang dimasukkan salah.

#### b. Implementasi Algoritma Antrian Proses

```

1. procedure TForm1.Button3Click(Sender: TObject);
2. begin
3.     qtampil.Close;
4.     qtampil.SQL.Clear;
5.     qtampil.SQL.Add('select faktur, kodevoucher, msisdn,
6.     hargajual from penjualan where faktur="'+
7.     string(etampil.Text) +'");
8.     qtampil.Open;
9.     qttotal.Close;
10.    qttotal.SQL.Clear;
11.    qttotal.SQL.Add('SELECT CONCAT("Total Pembelian Rp. ", ' +
12.    'REPLACE(FORMAT(SUM(hargajual),0),"",",",")", "-") AS total'
13.    +
14.    ' FROM penjualan WHERE faktur="'+ string(etampil.Text)
15.    +'");
16.    qttotal.Open;
17.    lbtotal.Caption := qttotal.FieldByName('total').AsString;
18.
19.
20. end;

```

**Gambar 4.29 Implementasi Algoritma antrian transaksi (preview)**

**Sumber : [Implementasi]**

Penjelasan implementasi Antrian Transaksi (Preview) Gambar 4.29 yaitu:

1. Baris 2-7 menjelaskan proses administrator memasukkan nomor faktur pada antrian proses. Nomor faktur tersebut menyeleksi transaksi sesuai dengan nomor faktur yang diinputkan
2. Baris 10-15 menjelaskan proses penggabungan transaksi yang ada pada nomor faktur dan diganti dengan menjumlahkan total transaksi sesuai dengan jumlah transaksi pada nomor faktur yang ditampilkan.

#### 4.3.4.3 Algoritma Aplikasi SMS Gateway

Algoritma proses yang ada pada aplikasi *SMS gateway* antara lain adalah proses *connect*, proses *SMS masuk*, proses *SMS keluar*. Implementasi algoritma *connect* akan direpresentasikan dalam bentuk *code* dengan bahasa pemrograman Delphi pada gambar 4.34.

```

1. procedure TForm1.CONNECTClick(Sender: TObject);
2. begin
3.     if connect.Caption = 'Disconnect' then
4.     begin
5.         connect.Caption := 'Connect';
6.         comport1.Close;
7.         mem1.Text := mem1.Text + #13#10 + 'Connection
8. Closed';
9.         timer1.Enabled := false;
10.    end
11.    else
12.    begin
13.        connect.Caption := 'Disconnect';
14.        mem1.Text := mem1.Text + #13#10 + 'Connecting to
15. modem';
16.        comport1.Open;
17.        comport1.WriteStr('AT+CMGF=1' + #13#10);
18.        timer1.Enabled := true;
19.    end;

```

**Gambar 4.30 Implementasi Algoritma SMS Gateway (connect)**

**Sumber : [Implementasi]**

Penjelasan implementasi *conect* Gambar 4.30 yaitu:

1. Baris 2-18 menjelaskan proses connect pada aplikasi *SMS gateway*. Pada saat proses connect berlangsung, memo 1 berisi SMS masuk dan memo 2 berisi SMS keluar yang dihasilkan oleh aplikasi.

#### 4.3.5 Implementasi Antarmuka

Antarmuka Aplikasi *Desktop* menggunakan nomor faktur sebagai antrian proses digunakan oleh sistem untuk berinteraksi dengan *user*, admin, serta server. Antarmuka perangkat lunak ini dibagi menjadi dua bagian, yaitu Antarmuka utama (aplikasi pulsa) dan Antarmuka administrator (kasir).

##### 4.3.5.1 Implementasi Antarmuka Aplikasi Pulsa

###### a. Antarmuka Home

Halaman home menampilkan form menu isi pulsa dan admin. Tombol isi pulsa mengarahkan *user* untuk melakukan pengisian pulsa. Sedangkan untuk tombol admin, diperuntukkan untuk operator yang tertuju pada form login untuk masuk pada aplikasi kasir. Gambar 4.31 akan menunjukkan implementasi tampilan antarmuka dari halaman utama yang mengacu pada perancangan Antarmuka masuk Sub Bab 4.2.2.1 bagian tampilan utama.



Gambar 4.31 Antarmuka Home

Sumber : [Implementasi]

b. Antarmuka Halaman XL

Halaman indosat menampilkan list voucher serta menampilkan kolom nomor untuk penginputan nomor yang akan diisi pulsa. Gambar 4.32 akan menunjukkan implementasi tampilan antarmuka dari halaman utama yang mengacu pada perancangan Antarmuka masuk Sub Bab 4.2.2.1 bagian tampilan Antarmuka xl.



**Gambar 4.32 Antarmuka XL**  
**Sumber : [Implementasi]**

c. Antarmuka Tampilan Nomor Faktur

Halaman nomor faktur menampilkan urutan transaksi sesuai dengan antrian. Satu nomor faktur menyimpan informasi tentang semua transaksi yang telah dilakukan. Gambar 4.33 akan menunjukkan implementasi tampilan antarmuka dari halaman utama yang mengacu pada perancangan Antarmuka masuk Sub Bab 4.2.3.1 bagian Antarmuka tampilan nomor faktur.



**Gambar 4.33 Antarmuka Tampilan Nomor Faktur**  
**Sumber : [Implementasi].**

#### 4.3.5.2 Implementasi Antarmuka Aplikasi Kasir

##### a. Antarmuka Login

Halaman login menampilkan form login untuk administrator untuk masuk pada halaman aplikasi kasir. Gambar 4.34 akan menunjukkan implementasi tampilan antarmuka dari halaman login yang mengacu pada perancangan Antarmuka masuk Sub Bab 4.2.2.2 bagian tampilan halaman login



**Gambar 4.34 Antarmuka Login**  
Sumber : [Implementasi]

##### b. Antarmuka Antrian Proses

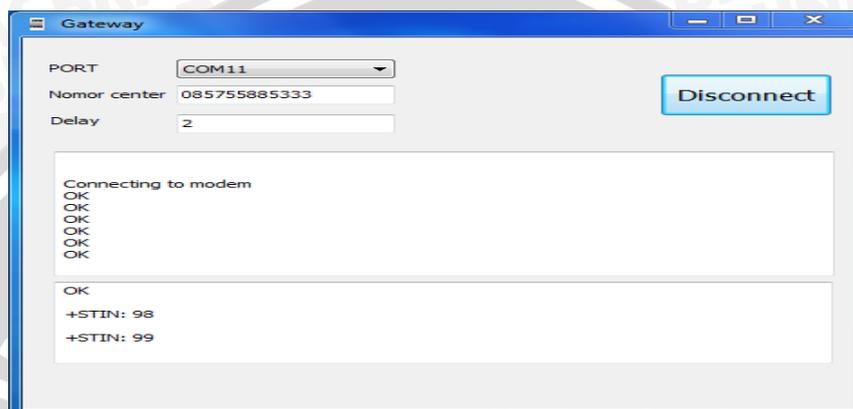
Halaman antrian proses menampilkan form antrian transaksi untuk memasukkan nomor faktur dari aplikasi utama. Halaman ini berfungsi sebagai tampilan untuk melihat jumlah transaksi dalam satu nomor faktur. Gambar 4.35 akan menunjukkan implementasi tampilan antarmuka dari halaman antrian proses yang mengacu pada perancangan Antarmuka masuk Sub Bab 4.2.2.2 bagian tampilan antarmuka antrian transaksi.



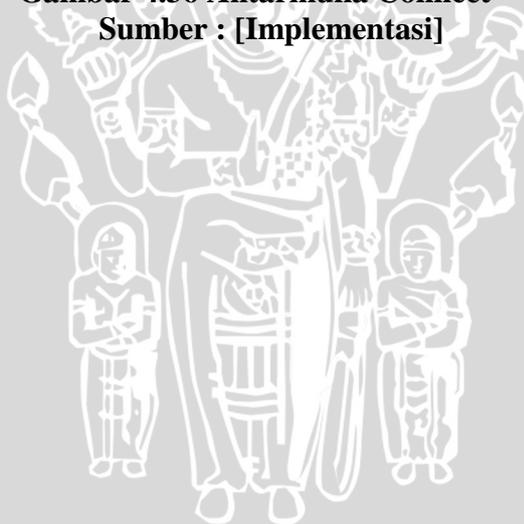
**Gambar 4.35 Antarmuka Antrian Proses**  
Sumber : [Implementasi]

#### 4.3.5.3 Implementasi Antarmuka Aplikasi Gateway

Halaman Gateway menampilkan label isi pesan berupa SMS keluar dan SMS masuk pada saat modem terkoneksi dengan aplikasi penjualan pulsa. Gambar 4.36 akan menunjukkan implementasi tampilan antarmuka dari halaman antrian proses yang mengacu pada perancangan antarmuka Sub Bab 4.2.3.2. bagian antarmuka halaman *connect* administrator.



**Gambar 4.36 Antarmuka Connect**  
**Sumber : [Implementasi]**



## BAB V

### PENGUJIAN DAN ANALISIS

Bab ini membahas mengenai tahapan pengujian dan analisis perangkat lunak Aplikasi Desktop penjualan pulsa dengan nomor faktur sebagai antrian proses. Proses pengujian dilakukan melalui tiga macam pengujian yaitu pengujian fungsional, pengujian non fungsional dengan menggunakan UAT (*User Acceptance Test*) serta pengujian performa. Pengujian fungsional digunakan untuk menguji kebenaran jalannya program sesuai dengan kebutuhan fungsional pengguna. Pengujian UAT (*User Acceptance Test*) dilakukan untuk mendapatkan data sejauh mana perangkat lunak yang dibangun telah memberikan kepuasan terhadap pengguna. Serta pengujian performa untuk mengetahui seberapa besar efektifitas dan efisiensi aplikasi apabila diterapkan untuk proses transaksi.

Proses analisis bertujuan untuk mendapatkan informasi sejauh mana perangkat lunak yang dibangun sudah sesuai dengan kebutuhan pengguna. Dalam artian aplikasi yang dibuat tidak terdapat *bug* atau *error* pada saat digunakan. Proses analisis mengacu pada dasar teori sesuai dengan hasil pengujian yang didapatkan. Analisis dilakukan terhadap hasil pengujian di setiap tahap pengujian. Proses analisis yang dilakukan meliputi analisis hasil pengujian fungsional dan pengujian UAT (*User Acceptance Test*).

#### 5.1 Pengujian

Proses pengujian dilakukan melalui tiga tahapan, yaitu pengujian fungsional, pengujian UAT (*User Acceptance Test*), serta pengujian performa terhadap aplikasi desktop penjualan pulsa dengan menggunakan nomor faktur sebagai antrian proses.

##### 5.1.1 Pengujian Fungsional

Pengujian validasi digunakan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan. *Item - item* yang telah dirumuskan dalam daftar kebutuhan dan merupakan hasil analisis kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian fungsional berfungsi untuk menemukan kesesuaian antara kerja sistem dengan daftar

kebutuhan yang telah dirancang sebelumnya. Pada skripsi ini dilakukan pengujian fungsional terhadap perangkat lunak *Aplikasi Desktop* penjualan pulsa dengan menggunakan *SMS Gateway* sebagai notifikasi pending. Kasus dan Hasil Pengujian Fungsional dijelaskan pada tabel 5.1 sampai 5.14.

a. Kasus Uji Menekan Tombol Isi Pulsa

**Tabel 5.1 Kasus Uji Menekan Tombol Isi Pulsa**

Nama Kasus Uji	Kasus Uji Menekan Tombol <i>Isi Pulsa</i>
Objek Uji	Kebutuhan Fungsional (SRS_001_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan ketika menekan tombol isi pulsa. Sistem akan memproses inputan dengan menampilkan tampilan pilihan operator
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menekan tombol <i>isi pulsa</i> .
Hasil yang Diharapkan	Aplikasi dapat melakukan proses eksekusi pada saat <i>user</i> menekan tombol isi pulsa
Hasil yang Didapatkan	Aplikasi dapat menyeleksi operator apa saja yang tersedia dan menampilkannya pada sistem
Status Validitas	Valid

**Sumber : [Pengujian dan Analisis]**

b. Kasus Uji Melakukan Pilihan Operator

**Tabel 5.2 Kasus Uji Melakukan Pilihan Operator**

Nama Kasus Uji	Kasus Uji Melakukan Pilihan Operator
Objek Uji	Kebutuhan Fungsional (SRS_001_02)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan dapat memilih operator sesuai dengan transaksi yang dilakukan.
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi.

		2. <i>User</i> menekan image operator.
Hasil yang Diharapkan		Aplikasi dapat melakukan pilihan operator sesuai dengan inputan transaksi.
Hasil yang Didapatkan		Aplikasi dapat menampilkan form operator terpilih dengan fitur-fitur di dalamnya.
Status Validitas		Valid

Sumber : [Pengujian dan Analisis]

c. Kasus Uji Menginputkan Nomor

**Tabel 5.3 Kasus Uji Menginputkan Nomor**

Nama Kasus Uji		Kasus Uji Menginputkan Nomor
Objek Uji		Kebutuhan Fungsional (SRS_001_03)
Tujuan Pengujian		Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan <i>user</i> dapat menginputkan nomor sesuai dengan transaksi yang dilakukan.
Prosedur Uji		1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menginputkan nomor telephone
Hasil yang Diharapkan		Aplikasi dapat menampilkan nomor telephone sesuai dengan inputan <i>user</i> .
Status Validitas		Valid

Sumber : [Pengujian dan Analisis]

d. Kasus Uji Melakukan Pilihan Voucher

**Tabel 5.4 Kasus Uji Untuk Melakukan Pilihan Voucher**

Nama Kasus Uji		Kasus Uji Melakukan Pilihan Voucher
Objek Uji		Kebutuhan Fungsional (SRS_001_04)
Tujuan Pengujian		Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan dapat memilih voucher sesuai dengan transaksi yang dilakukan.
Prosedur Uji		1. Sistem dijalankan saat program dieksekusi.

		2. <i>User</i> menekan nominal voucher
Hasil yang Diharapkan	yang	Aplikasi dapat memilih voucher sesuai dengan inputan <i>user</i> .
Hasil yang Didapatkan	yang	Aplikasi dapat menampilkan nominal voucher terpilih pada <i>field</i> keterangan yang berfungsi sebagai <i>note</i> terhadap transaksi yang dilakukan
Status Validitas		Valid

Sumber : [Pengujian dan Analisis]

e. Kasus Uji Melakukan Tambah Transaksi

**Tabel 5.5 Kasus Uji Untuk Melakukan Tambah Transaksi**

Nama Kasus Uji		Kasus Uji Melakukan Tambah Transaksi
Objek Uji		Kebutuhan Fungsional (SRS_001_05)
Tujuan Pengujian		Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan <i>user</i> dapat melakukan tambah transaksi dengan pilihan operator yang sama ataupun berbeda.
Prosedur Uji		1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menekan tombol tambah
Hasil yang Diharapkan	yang	Aplikasi dapat melakukan tambah transaksi sesuai dengan inputan <i>user</i> dengan cara menekan tombol tambah.
Hasil yang Didapatkan	yang	Aplikasi dapat menampilkan halaman pilihan operator setelah <i>user</i> menekan tombol tambah pada halaman provider terpilih.
Status Validitas		Valid

Sumber : [Pengujian dan Analisis]

## f. Kasus Uji Melakukan Hapus Transaksi

**Tabel 5.6 Kasus Uji Untuk Melakukan Hapus Transaksi**

Nama Kasus Uji	Kasus Uji Melakukan Hapus Transaksi
Objek Uji	Kebutuhan Fungsional (SRS_001_06)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan <i>user</i> dapat melakukan hapus transaksi pada transaksi yang dilakukan
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menekan tombol hapus
Hasil yang Diharapkan	Aplikasi dapat melakukan hapus transaksi sesuai dengan inputan <i>user</i> dengan cara menekan tombol hapus.
Hasil yang Didapatkan	Aplikasi dapat menampilkan sisa transaksi yang ada apabila transaksi yang dilakukan <i>user</i> lebih dari satu kali proses transaksi
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

## g. Kasus Uji Menampilkan Nomor Faktur

**Tabel 5.7 Kasus Uji Menampilkan Nomor Faktur**

Nama Kasus Uji	Kasus Uji Menampilkan Nomor Faktur
Objek Uji	Kebutuhan Fungsional (SRS_001_07)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan dapat melakukan cetak nomor faktur dari transaksi yang telah dilakukan
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menekan tombol proses
Hasil yang Diharapkan	Aplikasi dapat menampilkan nomor faktur sesuai dengan nomor urut transaksi yang dilakukan
Hasil yang Didapatkan	Aplikasi dapat menyeleksi transaksi berdasarkan

Didapatkan	nomor faktur apabila melakukan transaksi lebih dari satu.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

h. Kasus Uji Login

**Tabel 5.8 Kasus Uji Melakukan Login**

Nama Kasus Uji	Kasus Uji Menampilkan Halaman Login
Objek Uji	Kebutuhan Fungsional (SRS_002_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat menampilkan halaman login dan administrator dapat masuk ke dalam sistem.
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. Sistem menampilkan halaman login</li> <li>3. Administrator menginputkan <i>username</i> dan <i>password</i></li> </ol>
Hasil yang Diharapkan	Administrator dapat masuk ke dalam aplikasi kasir dengan menggunakan <i>username</i> dan <i>password</i> sesuai id tertentu.
Hasil yang Didapatkan	Aplikasi dapat berjalan dan administrator dapat mengakses halaman login dan masuk pada aplikasi kasir.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

i. Kasus Uji Melakukan Check Antrian Proses

**Tabel 5.9 Kasus Uji Melakukan Check Antrian Proses**

Nama Kasus Uji	Kasus Uji Melakukan Check Antrian Proses
Objek Uji	Kebutuhan Fungsional (SRS_002_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi

	<p>dapat berjalan dan administrator dapat mengecek status transaksi dan memasukkan nomor faktur yang didapat dari aplikasi utama.</p>
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. <i>administrator</i> menekan tombol antrian proses</li> <li>3. <i>administrator</i> menginputkan nomor faktur</li> </ol>
Hasil yang Diharapkan	Aplikasi dapat menampilkan transaksi sesuai dengan inputan nomor faktur oleh administrator
Hasil yang Didapatkan	Aplikasi dapat menampilkan banyaknya jumlah nominal transaksi berdasarkan nomor faktur yang diinputkan.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

j. Kasus Uji Melakukan Check Data Transaksi

**Tabel 5.10 Kasus Uji Untuk Melakukan Check Data Transaksi**

Nama Kasus Uji	Kasus Uji Melakukan Check Data Transaksi
Objek Uji	Kebutuhan Fungsional (SRS_002_02)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan administrator dapat mengecek data transaksi berdasarkan waktu transaksi dan status transaksi.
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. <i>administrator</i> menekan tombol data transaksi</li> <li>3. <i>administrator</i> check data transaksi</li> </ol>
Hasil yang Diharapkan	Aplikasi menampilkan transaksi berdasarkan inputan jangka waktu yang dilakukan oleh administrator
Hasil yang Didapatkan	Aplikasi menampilkan status transaksi berdasarkan inputan yang dilakukan administrator
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

## k. Kasus Uji Melakukan Setting

**Tabel 5.11 Kasus uji melakukan setting**

Nama Kasus Uji	Kasus Uji Melakukan Setting
Objek Uji	Kebutuhan Fungsional (SRS_002_03)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan administrator dapat mensetting segala kebutuhan aplikasi berdasarkan status operator, nominal voucher, harga beli, dan harga jual
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. <i>administrator</i> menekan tombol setting</li> <li>3. <i>administrator</i> mensetting status provider dan nominal yang tersedia</li> <li>4. <i>administrator</i> mensetting harga jual dan beli voucher</li> </ol>
Hasil yang Diharapkan	Aplikasi dapat mensetting status operator dan voucher sesuai dengan inputan yang dilakukan administrator
Hasil yang Didapatkan	Aplikasi dapat mensetting harga beli dan harga jual voucher sehingga dapat diketahui laba dalam transaksi yang diproses
Status Validitas	Valid

**Sumber : [Pengujian dan Analisis]**

## l. Kasus Uji Mengaktifkan SMS Gateway

**Tabel 5.12 Kasus uji mengaktifkan SMS gateway**

Nama Kasus Uji	Kasus Uji Mengaktifkan SMS Gateway
Objek Uji	Kebutuhan Fungsional (SRS_003_01)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan SMS gateway dapat terkoneksi dengan aplikasi.
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem SMS gateway dijalankan saat program</li> </ol>

		utama (pulsa dan kasir) dieksekusi. 2. <i>administrator</i> menekan tombol <i>connect</i>
Hasil yang Diharapkan	yang	Aplikasi gateway dapat terkoneksi dengan aplikasi utama
Hasil yang Didapatkan	yang	Aplikasi gateway dapat terkoneksi dengan sistem utama
Status Validitas		Valid

Sumber : [Pengujian dan Analisis]

m. Kasus Uji Mengirim SMS (Sms ke Server dan Sms Pending)

**Tabel 5.13 Kasus Uji Mengirim Sms**

Nama Kasus Uji		Kasus Uji Mengirim SMS
Objek Uji		Kebutuhan Fungsional (SRS_003_02)
Tujuan Pengujian		Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan aplikasi dapat mengirim SMS kepada server
Prosedur Uji		<ol style="list-style-type: none"> <li>1. Sistem SMS gateway dijalankan saat program utama (pulsa dan kasir) dieksekusi</li> <li>2. administrator menekan tombol <i>proses</i> pada aplikasi kasir setelah menginputkan nomor faktur</li> <li>3. Administrator mensetting delay sms pada aplikasi gateway untuk proses berapa lama aplikasi mengirimkan notifikasi apabila terjadi pending</li> </ol>
Hasil yang Diharapkan	yang	Aplikasi gateway dapat terkoneksi dengan aplikasi utama dan dapat mengirim SMS menuju server sesuai dengan format dan mengirimkan notifikasi pending apabila transaksi yang dilakukan melebihi waktu yang ditetapkan
Hasil yang Didapatkan	yang	<ol style="list-style-type: none"> <li>1. Aplikasi <i>SMS Gateway</i> dapat terhubung dengan PC dan utama (aplikasi pulsa dan aplikasi kasir)</li> <li>2. <i>SMS Gateway</i> dapat mengirimkan sms ke server</li> </ol>

	sesuai dengan format transaksi berdasarkan kodevoucher dan nomor telepon yang telah dipilih
	3. Aplikasi <i>SMS Gateway</i> dapat mengirimkan sms berupa notifikasi kepada pembeli berdasarkan waktu delay yang telah ditentukan apabila terjadi pending atau transaksi gagal.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

n. Kasus Uji Menerima Sms

**Tabel 5.14 Kasus Uji Menerima Sms**

Nama Kasus Uji	Kasus Uji Menerima <i>SMS</i>
Objek Uji	Kebutuhan Fungsional (SRS_003_03)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat berjalan dan aplikasi dapat menerima <i>SMS</i> dari server
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem <i>SMS gateway</i> dijalankan saat program utama (pulsa dan kasir) dieksekusi.</li> <li>2. <i>administrator</i> menekan tombol <i>proses</i> pada aplikasi kasir setelah menginputkan nomor faktur</li> </ol>
Hasil yang Diharapkan	Aplikasi gateway dapat terkoneksi dengan aplikasi utama dan dapat menerima <i>SMS</i> dari server sebagai laporan transaksi berhasil
Hasil yang Didapatkan	Aplikasi gateway dapat terkoneksi dengan sistem utama dan dapat menerima <i>SMS</i> dari server sebagai balasan transaksi telah berhasil.
Status Validitas	Valid

### 5.1.2 Pengujian UAT (*User Acceptance Testing*)

Pengujian UAT digunakan untuk mendapatkan konfirmasi melalui *trial* atau *review* dari pihak yang terkait sesuai dengan *requirement* yang telah disetujui sebelumnya. Dari pengujian ini diharapkan dapat diketahui apakah sistem yang dibangun telah sesuai dengan yang diharapkan *user*.

Tahap pengujian UAT ini terdiri dari 2 faktor uji, yaitu *ease of use* dan *perceive of usefulness*. Berikut tabel hasil UAT yang diujikan kepada 45 *user*, dan 5 administrator yang dalam hal ini bertindak sebagai counter. Pada pengujian UAT yang diajukan, parameter yang digunakan menghasilkan skor 1 sampai dengan skor 5 dengan keterangan skor 1 adalah kurang, 2 adalah cukup, 3 adalah sangat cukup, 4 adalah baik, dan 5 mempunyai keterangan baik sekali.

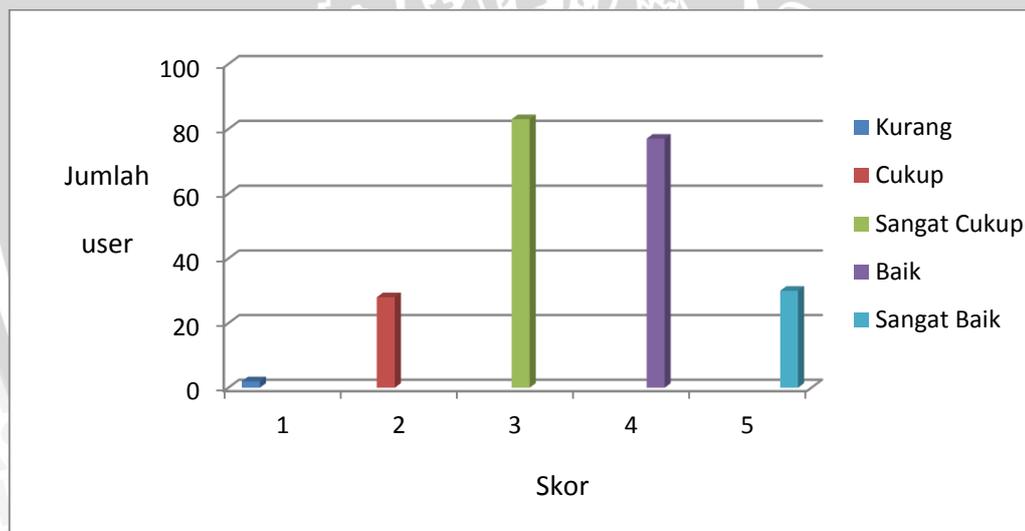
**Tabel 5.15 Tabel Pengujian UAT terhadap 45 *user***

NAMA USER	PERCEIVE OF USEFULNESS					AVERAGE	EASE OF USE					AVERAGE
	Q1	Q2	Q3	Q4	Q5		Q1	Q2	Q3	Q4	Q5	
VINA ALFIANI	5	5	3	3	4	4,0	2	2	3	1	3	2,2
IDZA ZULFILATIN	5	4	4	3	2	3,6	2	1	3	3	4	2,6
SHAFIRA NUR	3	2	3	4	4	3,2	2	3	4	4	3	3,2
JENNY CHRISMAWATI	4	3	4	5	3	3,8	2	2	4	3	2	2,6
DWI PUSPITASARI	3	4	3	2	3	3,0	2	2	3	3	2	2,4
FERYZAL KASANDE	5	3	3	4	2	3,4	3	2	3	2	2	2,4
PUTRI MUTIARA VAQI	5	4	3	4	3	3,8	1	2	3	3	2	2,2
ILHAM NUR HUDA	5	4	3	2	3	3,4	3	2	3	2	2	2,4
CHAINUR RACHMAN	5	4	2	3	3	3,4	2	3	4	4	3	3,2
NOVANZI RISKI	4	3	2	4	2	3,0	3	2	5	4	3	3,4
DWI YOSSY	5	3	3	2	1	2,8	3	2	3	4	2	2,8
TITIK MULASIH	4	4	3	2	3	3,2	3	2	3	2	2	2,4
KHURNIAWATI	4	4	3	2	2	3,0	3	3	3	3	2	2,8
RESKY RACHMAN	4	3	3	3	2	3,0	3	4	2	3	3	3,0
LANY ISMI AROFAH	5	4	3	3	2	3,4	3	2	2	3	3	2,6
FEBRINA T.R	4	1	2	3	4	2,8	1	2	2	3	3	2,2
A. FANANI	4	3	3	2	3	3,0	2	3	2	3	3	2,6
RINA ZULKARNAIN	4	3	4	2	5	3,6	2	3	4	3	2	2,8
NATASYA QUAMILA	4	3	4	2	3	3,2	3	2	2	3	3	2,6
STEVEN EFENDI	4	3	4	3	2	3,2	2	3	3	2	3	2,6
MIFTA SARI R.	4	3	4	3	2	3,2	3	4	3	5	3	3,6
ACHMAD YUDI A.	5	4	4	4	2	3,8	3	4	4	3	3	3,4
YAHYA EFENDI	5	4	4	3	4	4,0	4	4	5	5	3	4,2
FIRMAWAN ARDIANSYAH	5	4	3	4	3	3,8	4	4	4	3	3	3,6
DARIA ANGGRAENI	5	4	4	4	4	4,2	4	4	4	5	4	4,2
ARDHI ARIMA	5	4	4	4	3	4,0	3	4	4	3	5	3,8
UMNIATUL MUNAWAROH	4	3	4	3	2	3,2	3	2	3	3	4	3,0
CITRA TRIMISWATI	4	4	3	5	3	3,8	3	3	2	3	3	2,8
DANI PRIBAWANTO	5	4	3	4	3	3,8	2	4	4	3	3	3,2
NASRUL ANNAFIQ	5	4	3	4	4	4,0	3	3	4	4	3	3,4
NIKEN KARTIKA	4	3	3	3	2	3,0	2	3	2	4	3	2,8
M. IMAM SAFTI	4	3	5	4	4	4,0	5	2	3	1	3	2,8
M. RISKI NOVIALDI	4	3	5	4	5	4,2	3	2	4	3	4	3,2
ONI FEBRIANTO	3	4	4	4	3	3,6	4	3	4	3	4	3,6
VALENSARI	5	3	3	3	5	3,8	4	4	3	3	3	3,4
DIKA ALAMSYAH	3	4	5	3	5	4,0	4	3	4	3	5	3,8
SATRIYA SAPUTRA	3	4	3	4	5	3,8	3	3	4	2	3	3,0
RAHMAT HIDAYAT	2	3	4	2	5	3,2	3	3	2	2	3	2,6
SIGIT PURNAMA	4	3	3	5	3	3,6	4	3	3	2	3	3,0
DYAH AYUNDA	3	2	3	4	5	3,4	3	4	2	4	2	3,0
TRIA FEBRIANI	3	2	3	4	3	3,0	3	2	3	3	3	2,8
RENI ISTIQOWATI	3	4	3	3	2	3,0	2	3	4	3	3	3,0
DICKY ARIEF PRATAMA	3	3	3	3	4	3,2	3	4	3	1	4	3,0
JONATA R.	4	3	4	3	4	3,6	4	4	4	3	3	3,6
<b>RATA-RATA</b>	<b>KEGUNAAN FUNGSI</b>					<b>3,48</b>	<b>KEMUDAHAN FUNGSI</b>					<b>3,00</b>

[Sumber : Lampiran]

Dari tabel 5.15 dapat disimpulkan bahwa skor yang diperoleh berbeda-beda tiap pengguna, begitu pula dengan jumlah dan rata-rata yang didapat. Jika dilihat dari segi kegunaan fungsi (*perceive usefulness*) nilai rata-rata maksimal dari 45 *user* adalah 4,2 dan nilai minimumnya 2,8 dengan rata-rata keseluruhan mencapai 3,48 dengan keterangan sangat cukup. Sedangkan Jika dilihat dari segi kemudahan fungsi (*ease of use*) nilai rata-rata maksimal dari 45 *user* adalah 4,2 dan nilai minimumnya 2,2 dengan rata-rata keseluruhan mencapai 3,0.

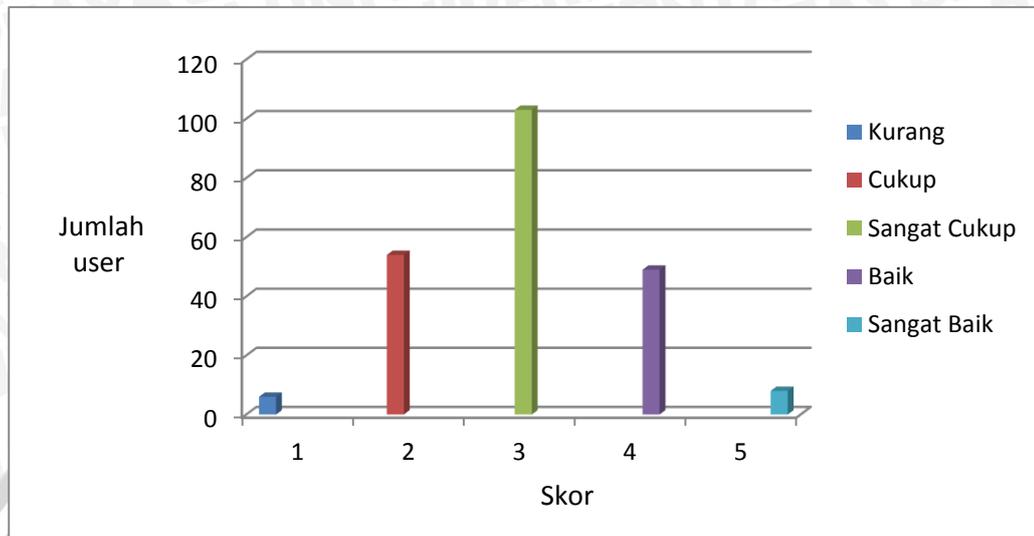
Rata-rata yang didapatkan dari pengujian kuisisioner diatas menandakan tingkat kemudahan dan kegunaan dari aplikasi yang telah dibuat memperoleh keterangan sangat cukup berdasarkan parameter yang digunakan pada form kuisisioner. Kekurangan yang didapat dari pengujian sitem yaitu kurangnya pemahaman pengguna tentang fitur-fitur yang ada dalam ssitem serta tampilan desktop yang kurang memudahkan pengguna karena kebanyakan pengguna masih awam terhadap proses bisnis yang ada pada aplikasi tersebut.



**Gambar 5.1 : Grafik *Perceive Of Use* Pengujian Terhadap 45 User**

Gambar 5.1 adalah grafik yang menjelaskan tentang hasil pengujian UAT dengan faktor uji *perceive of use* dari 45 *user* pengguna aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses. Dari grafik diatas didapatkan jumlah jawaban yang berbeda setiap pengguna. Skor yang didapatkan

pada pengujian UAT ini berdasarkan keterangan skor yang terlampir pada lampiran kuisioner.



**Gambar 5.2 : Grafik *Ease of Use* Pengujian Terhadap 45 User**

Gambar 5.2 adalah grafik yang menjelaskan tentang hasil pengujian UAT dengan faktor uji *ease use* dari 45 user pengguna aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses. Dari grafik diatas didapat jumlah jawaban pengguna berdasarkan skor dengan keterangan yang terlampir pada lampiran kuisioner.

**Tabel 5.16 Tabel Pengujian UAT terhadap 5 Administrator**

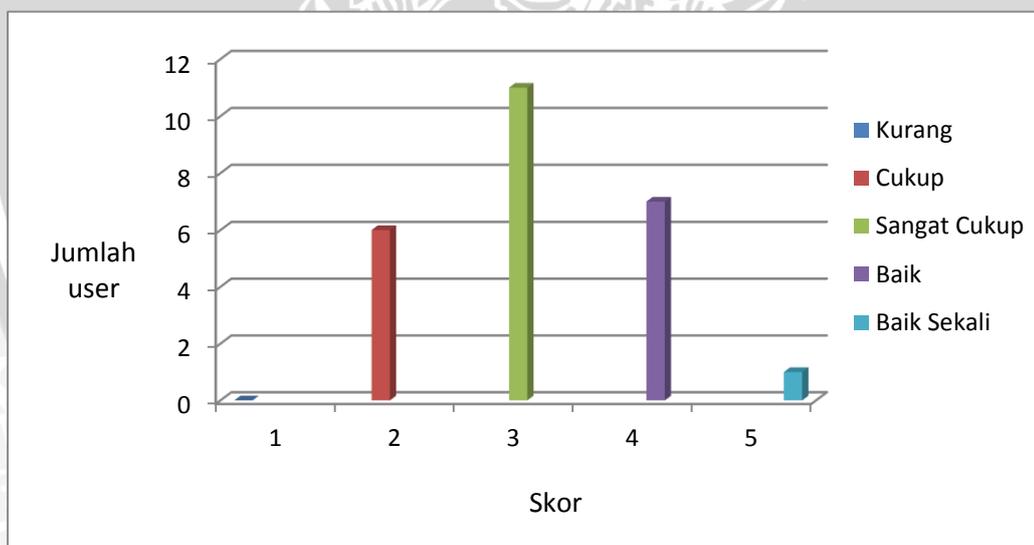
NAMA COUNTER	PERCEIVE OF USEFULNESS					AVERAGE	EASE OF USE					AVERAGE
	Q1	Q2	Q3	Q4	Q5		Q1	Q2	Q3	Q4	Q5	
FIRST CELL	5	4	4	3	2	3,6	4	2	3	2	3	2,8
KEDUNG EMAS CELL	5	4	3	1	2	3	2	4	3	3	4	3,2
BINTANG 7 CELL	5	4	3	3	1	3,2	2	3	4	3	4	3,2
SUMBER ABADI CELL	5	4	4	3	2	3,6	2	3	3	2	3	2,6
DWI PUSPITASARI	5	4	4	3	3	3,8	4	4	3	3	5	3,8
<b>RATA-RATA</b>	<b>KEGUNAAN FUNGSI</b>					<b>3,44</b>	<b>KEMUDAHAN FUNGSI</b>					<b>3,12</b>

[Sumber : Lampiran]

Dari tabel 5.13 dapat disimpulkan bahwa skor yang diperoleh berbeda-beda tiap administrator, begitu pula dengan jumlah dan rata-rata yang didapat. Jika dilihat dari segi kegunaan fungsi (*percieve usefulness*) nilai rata-rata

maksimal dari 5 administrator adalah 3,8 dan nilai minimumnya 3,0 dengan rata-rata keseluruhan mencapai 3,44 dengan keterangan sangat cukup. Sedangkan Jika dilihat dari segi kemudahan fungsi (*ease of use*) nilai rata-rata maksimal dari 5 administrator adalah 3,8 dan nilai minimumnya 2,8 dengan rata-rata keseluruhan mencapai 3,12.

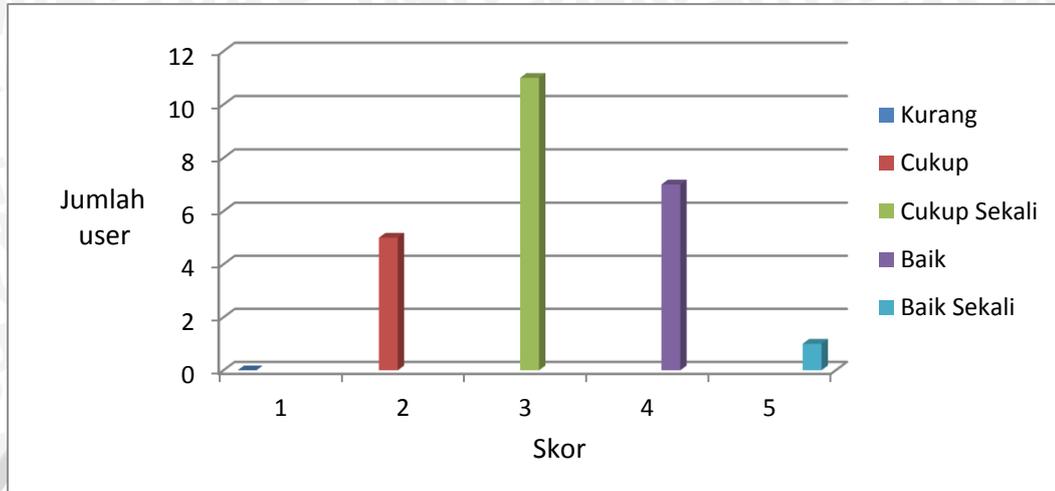
Rata-rata yang didapatkan dari pengujian kuisisioner diatas menandakan tingkat kemudahan dan kegunaan dari aplikasi yang telah dibuat memperoleh keterangan sangat cukup berdasarkan parameter yang digunakan pada form kuisisioner. Kekurangan yang didapat dari pengujian sitem ini yaitu tingkat kebutuhan pada masing-masing counter berbeda sehingga fitur-fitur yang digunakan belum tentu cocok pada setiap counter. Dari tingkat kebutuhan yang berbeda tersebut mempengaruhi perolehan skor yang ada. Selain itu pengujian UAT ini menggunakan contoh counter pada tingkat yang berbeda. Dari counter tingkat bawah sampai counter tingkat atas sehingga mempengaruhi jawaban dari masing-masing counter.



**Gambar 5.3 : Grafik *Perceive Of Use* Pengujian Terhadap 5 Administrator**

Gambar 5.3 adalah grafik yang menjelaskan tentang hasil pengujian UAT dengan faktor uji *ease use* dari 5 administrator pengguna aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses. Dari grafik diatas

didapat jumlah jawaban pengguna berdasarkan skor dengan keterangan yang terlampir pada lampiran kuisioner.



**Gambar 5.4 : Grafik *Ease Of Use* Pengujian Terhadap 5 Administrator**

### 5.1.3 Pengujian Performa

Pengujian performa digunakan untuk mengetahui apakah sistem yang dibangun dapat melakukan proses pengisian pulsa dengan cepat dengan sistem antrian yang cepat pula, sehingga dapat memberikan kenyamanan kepada pembeli. Selain itu pengujian ini dilakukan dengan cara bertahap dimana dilakukan pengujian pada satu transaksi terlebih dahulu kemudian melakukan pengujian secara *multi*. Pada pengujian satu transaksi akan dilakukan dengan tujuan untuk mencari rentang waktu terbaik pada masing-masing proses yang terjadi. Sedangkan untuk pengujian secara *multi* dilakukan secara bertahap dengan melakukan 5 transaksi, 10 transaksi, dan 20 transaksi sehingga didapatkan performa aplikasi yang dilihat dari persentase keberhasilan dan waktu tunggu selama proses mulai berjalan sampai proses selesai. Pada skripsi ini dilakukan pengujian performa terhadap perangkat lunak aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses.

Tabel 5.17 Antrian 1 transaksi sebelum menggunakan aplikasi

<i>Id Antrian</i>	<b>Waktu Datang (d)</b>	<b>Waktu Bayar</b>	<b>Waktu Proses (m)</b>	<b>Waktu Tunggu (d-m) (menit)</b>
1	07.55	07.57	07.59	2
<b>Rata-rata Waktu Tunggu (Menit)</b>				<b>2</b>

Dari tabel 5.17 dapat disimpulkan bahwa waktu tunggu pembeli pada pengujian 1 transaksi memperoleh selisih waktu selama 2 menit dengan rata-rata waktu tunggu pembeli sebelum menggunakan aplikasi ini digunakan adalah 2 menit.

Tabel 5.18 Antrian 5 transaksi sebelum menggunakan aplikasi

<i>Id Antrian</i>	<b>Waktu Datang (d)</b>	<b>Waktu Bayar</b>	<b>Waktu Proses (m)</b>	<b>Waktu Tunggu (d-m) (menit)</b>
1	07.55	07.57	07.59	4
2	07.55	07.57	08:00	5
3	07.56	07.57	08:01	5
4	07.55	07.57	08:02	5
5	07.57	07:57	08:03	6
<b>Rata-rata Waktu Tunggu (Menit)</b>				<b>5,8</b>

Dari tabel 5.18 dapat disimpulkan bahwa waktu tunggu pembeli pada pengujian 5 transaksi bergantung pada berapa banyak antrian yang ada dengan menghasilkan waktu antrian dari antrian ke-1 dengan antrian ke-4 sebanyak 2 menit. Pada pengujian 4 transaksi terjadi selang waktu antrian ke-1 dengan antrian ke-5 sebanyak 4-6 menit dengan rata-rata waktu tunggu keseluruhan sebelum menggunakan aplikasi ini digunakan adalah 5,8 menit.

Tabel 5.19 Antrian 10 transaksi sebelum menggunakan aplikasi

<i>Id Antrian</i>	<b>Waktu Datang (d)</b>	<b>Waktu Bayar</b>	<b>Waktu Proses (m)</b>	<b>Waktu Tunggu (d-m) (menit)</b>
1	07.55	07.59	08.00	5
2	07.55	07.59	08.00	5
3	07.56	07.59	08.01	5
4	07.56	07.59	08.02	6
5	07.57	07.59	08.03	6
6	07.57	07.59	08.03	6
7	07.58	07.59	08.04	6
8	07.58	07.59	08.04	6

9	07.59	07.59	08.06	7
10	07.59	07.59	08.06	7
<b>Rata-rata Waktu Tunggu (Menit)</b>				<b>5,9</b>

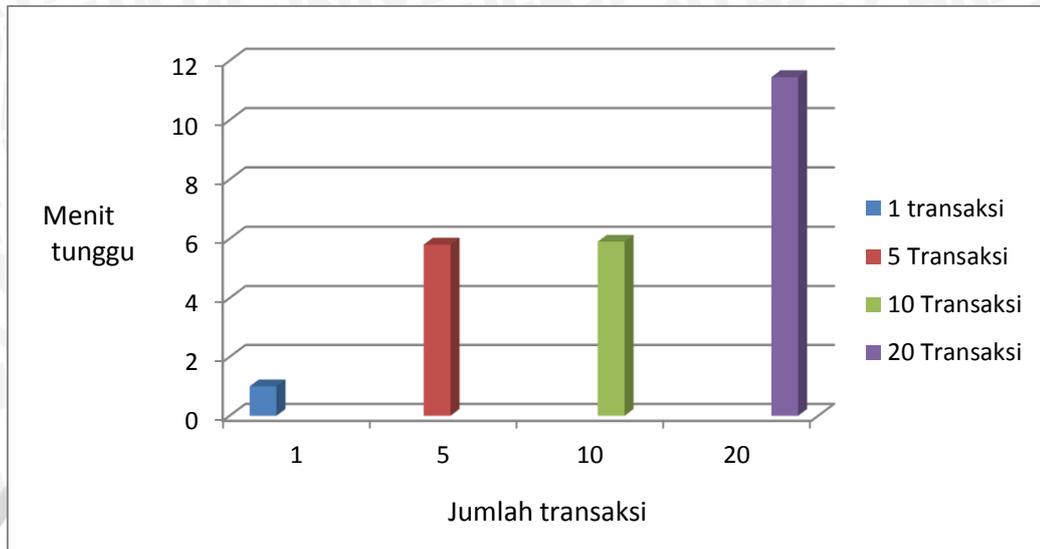
Dari tabel 5.19 dapat disimpulkan bahwa waktu tunggu pembeli pada pengujian 10 transaksi bergantung pada berapa banyak antrian yang ada dengan menghasilkan waktu antrian dari antrian ke-1 dengan antrian ke-10 sebanyak 4 menit. Pada pengujian 10 transaksi terjadi selang waktu antrian ke-1 dengan antrian ke-10 sebanyak 5-7 menit dengan rata-rata waktu tunggu keseluruhan sebelum menggunakan aplikasi ini digunakan adalah 5,9 menit.

**Tabel 5.20 Antrian 20 transaksi sebelum menggunakan aplikasi**

<b>Id Antrian</b>	<b>Waktu Datang (d)</b>	<b>Waktu Bayar</b>	<b>Waktu Proses (m)</b>	<b>Waktu Tunggu (d-m) (menit)</b>
1	07.55	08:04	08:05	10
2	07.55	08:04	08:05	10
3	07.56	08:04	08:06	10
4	07.56	08:04	08:06	10
5	07.57	08:04	08:07	10
6	07.57	08:04	08:07	10
7	07.58	08:04	08:09	11
8	07.58	08:04	08:09	11
9	07.59	08:04	08:11	12
10	07.59	08:04	08:11	12
11	08.00	08:04	08:12	12
12	08.00	08:04	08:12	12
13	08.01	08:04	08:13	12
14	08.01	08:04	08:13	12
15	08.02	08:04	08:14	12
16	08.02	08:04	08:14	12
17	08.03	08:04	08:15	12
18	08.03	08:04	08:16	13
19	08.04	08:04	08:16	13
20	08.04	08:04	08:16	13
<b>Rata-rata Waktu Tunggu (Menit)</b>				<b>11,45</b>

Dari tabel 5.20 dapat disimpulkan bahwa waktu tunggu pembeli pada pengujian 20 transaksi bergantung pada berapa banyak antrian yang ada dengan menghasilkan waktu antrian dari antrian ke-1 dengan antrian ke-20 sebanyak 9 menit. Pada pengujian 20 transaksi terjadi selang waktu antrian ke-1 dengan antrian

ke-20 sebanyak 10-13 menit dengan rata-rata waktu tunggu keseluruhan sebelum menggunakan aplikasi ini digunakan adalah 11,45 menit.



**Gambar 5.5 : Grafik Waktu Tunggu Pembeli Sebelum menggunakan Aplikasi**

Gambar 5.5 merupakan grafik yang menjelaskan tentang hasil pengujian waktu tunggu pembeli sebelum penggunaan aplikasi dengan pengujian menggunakan 1 transaksi, 5 transaksi, 10 transaksi, dan 20 transaksi. Dapat disimpulkan bahwa Dapat disimpulkan bahwa penumpukan waktu tunggu tertinggi terjadi pada pengujian 20 transaksi dengan waktu tunggu pembeli mencapai 11,45 menit.

**Tabel 5.21 antrian 1 transaksi setelah menggunakan aplikasi**

<i>Id Antrian</i>	<b>Waktu Datang (d)</b>	<b>Waktu Bayar</b>	<b>Waktu Proses (m)</b>	<b>Waktu Tunggu (d-m) (menit)</b>
1	07.55	07.56	07.56	1 menit
<b>Rata-rata Waktu Tunggu (Menit)</b>				<b>1</b>

Dari tabel 5.21 dapat disimpulkan bahwa waktu tunggu pembeli pada pengujian 1 transaksi memperoleh selisih waktu selama 1 menit dengan rata-rata waktu tunggu pembeli setelah menggunakan aplikasi ini adalah 1 menit.

Tabel 5.22 Antrian 5 transaksi setelah menggunakan aplikasi

<i>Id Antrian</i>	<b>Waktu Datang (d)</b>	<b>Waktu Bayar</b>	<b>Waktu Proses (m)</b>	<b>Waktu Tunggu (d-m) (menit)</b>
1	07.55	07.55	07.56	1
2	07.55	07.55	07.56	1
3	07.55	07.55	07.56	1
4	07.55	07.55	07.56	1
5	07.55	07.55	07.56	1
<b>Rata-rata Waktu Tunggu (Menit)</b>				<b>1</b>

Dari tabel 5.22 dapat disimpulkan bahwa waktu tunggu pembeli pada pengujian 5 transaksi dengan menggunakan aplikasi desktop dengan nomor faktur sebagai antrian proses tidak tergantung pada berapa banyak antrian pada transaksi yang ada. Waktu tunggu proses transaksi (waktu datang dan waktu bayar) adalah sama karena 5 transaksi tersebut tercatat dalam 1 nomor faktur sehingga data langsung diproses. Rata-rata waktu tunggu pembeli dalam 5 transaksi setelah menggunakan aplikasi ini adalah 1 menit.

Tabel 5.23 Antrian 10 transaksi setelah menggunakan aplikasi

<i>Id Antrian</i>	<b>Waktu Datang (d)</b>	<b>Waktu Bayar</b>	<b>Waktu Proses (m)</b>	<b>Waktu Tunggu (d-m) (menit)</b>
1	07.55	07.55	07.57	2
2	07.55	07.55	07.57	2
3	07.55	07.55	07.57	2
4	07.55	07.55	07.57	2
5	07.55	07.55	07.57	2
6	07.55	07.55	07.57	2
7	07.55	07.55	07.57	2
8	07.55	07.55	07.57	2
9	07.55	07.55	07.57	2
10	07.55	07.55	07.57	2
<b>Rata-rata Waktu Tunggu (Menit)</b>				<b>2</b>

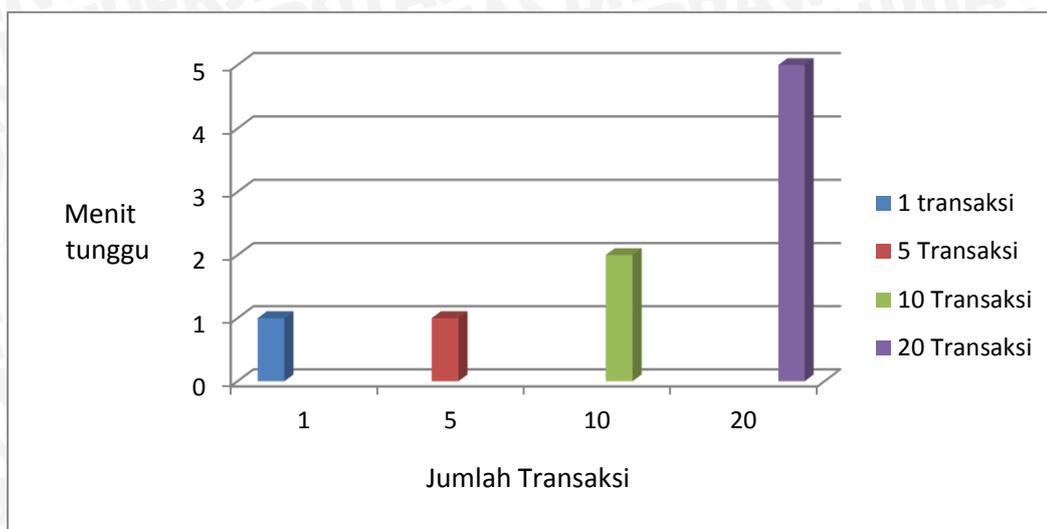
Dari tabel 5.23 dapat disimpulkan bahwa waktu tunggu pembeli pada pengujian 10 transaksi dengan menggunakan aplikasi desktop dengan nomor faktur sebagai antrian proses tidak tergantung pada berapa banyak antrian pada transaksi yang ada. Waktu tunggu proses transaksi (waktu datang dan waktu

bayar) adalah sama karena 10 transaksi tersebut tercatat dalam 1 nomor faktur sehingga data langsung diproses. Rata-rata waktu tunggu pembeli dalam 10 transaksi setelah menggunakan aplikasi ini adalah 2 menit.

**Tabel 5.24 Antrian 20 transaksi setelah menggunakan aplikasi**

<i><b>Id Antrian</b></i>	<b>Waktu Datang (d)</b>	<b>Waktu Bayar</b>	<b>Waktu Proses (m)</b>	<b>Waktu Tunggu (d-m) (menit)</b>
1	07.55	07.58	08:00	5
2	07.55	07.58	08:00	5
3	07.55	07.58	08:00	5
4	07.55	07.58	08:00	5
5	07.55	07.58	08:00	5
6	07.55	07.58	08:00	5
7	07.55	07.58	08:00	5
8	07.55	07.58	08:00	5
9	07.55	07.58	08:00	5
10	07.55	07.58	08:00	5
11	07.55	07.58	08:00	5
12	07.55	07.58	08:00	5
13	07.55	07.58	08:00	5
14	07.55	07.58	08:00	5
15	07.55	07.58	08:00	5
16	07.55	07.58	08:00	5
17	07.55	07.58	08:00	5
18	07.55	07.58	08:00	5
19	07.55	07.58	08:00	5
20	07.55	07.58	08:00	5
<b>Rata-rata Waktu Tunggu (Menit)</b>				<b>5</b>

Dari tabel 5.24 dapat disimpulkan bahwa waktu tunggu pembeli pada pengujian 20 transaksi dengan menggunakan aplikasi desktop dengan nomor faktur sebagai antrian proses tidak tergantung pada berapa banyak antrian pada transaksi yang ada. Waktu tunggu proses transaksi (waktu datang dan waktu bayar) adalah sama karena 20 transaksi tersebut tercatat dalam 1 nomor faktur sehingga data langsung diproses. Rata-rata waktu tunggu pembeli dalam 20 transaksi setelah menggunakan aplikasi ini adalah 5 menit.



**Gambar 5.6 : Grafik Waktu Tunggu Pembeli Setelah Menggunakan Aplikasi**

Gambar 5.6 merupakan grafik yang menjelaskan tentang hasil pengujian waktu tunggu pembeli setelah penggunaan aplikasi dengan pengujian menggunakan 1 transaksi, 5 transaksi, 10 transaksi, dan 20 transaksi. Dapat disimpulkan bahwa penumpukan waktu tunggu tertinggi terjadi pada pengujian 20 transaksi dengan waktu tunggu pembeli mencapai 5 menit.

## 5.2 Analisis

Proses analisis bertujuan untuk mendapatkan informasi sejauh mana perangkat lunak yang dibangun sudah sesuai dengan kebutuhan pengguna. Proses analisis mengacu pada dasar teori sesuai dengan hasil pengujian yang didapatkan. Analisis dilakukan terhadap hasil pengujian di setiap tahap pengujian. Proses analisis yang dilakukan meliputi analisis hasil pengujian black box dan analisis hasil pengujian UAT (*User Acceptance Testing*).

### 5.2.1 Analisis Hasil Pengujian Fungsional

Proses analisis terhadap hasil pengujian fungsional dilakukan dengan cara menguji segala fitur-fitur yang ada dan fungsi pada perangkat lunak apakah telah sesuai dengan kebutuhan pengguna. Berdasarkan hasil pengujian fungsional dapat disimpulkan bahwa implementasi dan fungsionalitas perangkat lunak *Aplikasi desktop penjualan pulsa* dengan menggunakan nomor faktur sebagai antrian

proses telah memenuhi kebutuhan yang telah dijabarkan pada tahap analisis kebutuhan.

Dalam tahap ini juga terdapat *bug* dan *error* dalam proses pengujian aplikasi desktop penjualan pulsa dengan *SMS gateway* sebagai notifikasi pending. Diantaranya sebagai berikut :

1. Error instalasi *component* pada *delphi*. Salah satu contoh, proses instalasi *component* modem pada *delphi* terkadang tidak dapat terdeteksi yang mengakibatkan modem tidak dapat terhubung dengan aplikasi.
2. Pada spesifikasi PC tertentu, aplikasi desktop tidak berjalan sebagaimana mestinya. Misalnya pada saat proses tambah transaksi, seharusnya voucher pulsa tidak tampil pada layar PC kecuali *user* mulai menginputkan nomor handphone. Hal ini sempat terjadi pada saat *user* melakukan proses tambah transaksi, voucher pulsa otomatis tampil tanpa harus menginputkan nomor tujuan tetapi proses pengisian pulsa berjalan sebagaimana mestinya.
3. Untuk perangkat modem yang digunakan dalam aplikasi ini adalah modem wavecom fastrack M1306B. Pada modem lain ditemukan bug yaitu pada aplikasi *SMS gateway*, perintah yang dihasilkan tidak dapat tampil pada layar atau kotak dialog.
4. Bug pada modem wavecom fastrack M1306B yaitu indikator *SMS* masuk dan delete berjalan sangat cepat yang menghasilkan tabrakan proses. Bug tersebut yang mengakibatkan terkadang *SMS* dari server tidak tersimpan pada database, yang mengakibatkan aplikasi tidak dapat mengirim *SMS* ke *user*.
5. Durasi delay pada aplikasi gateway dimulai pada saat pembeli mengakses aplikasi pulsa sampai mendapatkan nomor faktur dari transaksi yang dilakukan.

### 5.2.2 Analisis Hasil Pengujian UAT (*User Acceptance Testing*)

Proses analisis terhadap Pengujian UAT (*User Acceptance Testing*) terdiri dari 2 faktor uji, yaitu *ease of use*, *perceive of usefulness*. Kedua pengujian tersebut diujikan kepada 2 pihak selaku admin dan *user* dengan menitikberatkan kepada kemudahan serta kegunaan sistem dalam menggunakan aplikasi yang telah disiapkan. Berdasarkan pengujian UAT (*User Acceptance Testing*) yang dalam

hal ini berupa kuisioner pada 45 *user* dan 5 administrator, didapat beberapa kesimpulan sebagai berikut :

1. Rata-rata keseluruhan pada pengujian UAT pada 45 *user* berdasarkan kegunaan fungsi (*percieve usefulness*) memiliki rata-rata sebesar 3,48 dan kemudahan fungsi (*easy of use*) sebesar 3,0.
2. Rata-rata diatas menandakan aplikasi yang dibangun bernilai sangat cukup berdasarkan acuan pada tabel kuisioner. Dengan rata-rata yang didapat pada pengujian diatas, aplikasi tersebut dapat diterima oleh *user* dari segi kegunaan fungsi maupun kemudahan dalam menjalankan aplikasi tersebut.
3. Rata-rata keseluruhan pada pengujian UAT pada 5 administator berdasarkan kegunaan fungsi (*percieve usefulness*) memiliki rata-rata sebesar 3,44 dan kemudahan fungsi (*easy of use*) sebesar 3,12.
4. Rata-rata diatas menandakan aplikasi yang dibangun bernilai sangat cukup berdasarkan acuan pada tabel kuisioner. Dengan rata-rata yang didapat pada pengujian diatas, aplikasi tersebut dapat diterima oleh *user* dari segi kegunaan fungsi maupun kemudahan dalam menjalankan aplikasi tersebut.

### 5.2.3 Analisis Hasil Pengujian Performa

Proses analisis terhadap hasil pengujian performa dilakukan dengan menghitung nilai rata-rata dari waktu tunggu pembeli. Berdasarkan hal tersebut maka dapat diambil kesimpulan sebagai berikut :

- a. Penggunaan aplikasi desktop dengan menggunakan nomor faktur sebagai antrian proses lebih efektif dan efisien jika dibandingkan dengan menggunakan sistem lama yang menggunakan sistem secara manual dalam proses transaksi.
- b. Berkurangnya waktu tunggu, berbanding terbalik dengan tingkat kenyamanan pembeli. Saat penggunaan aplikasi desktop, rata-rata waktu tunggu pembeli berkurang, dengan tingkat kenyamanan yang meningkat.
- c. Perbandingan pengujian pada 5 transaksi sebelum memakai aplikasi memiliki rata-rata waktu tunggu 5,8 menit dan setelah menggunakan

aplikasi desktop dengan nomor faktur sebagai antrian proses memiliki waktu tunggu rata-tara sebesar 1 menit.

- d. Perbandingan pengujian pada 10 transaksi sebelum memakai aplikasi memiliki rata-rata waktu tunggu 5,9 menit dan setelah menggunakan aplikasi desktop dengan nomor faktur sebagai antrian proses memiliki waktu tunggu rata-tara sebesar 1 menit.
- e. Penumpukan waktu tunggu tertinggi pada sistem lama terjadi pada pengujian 20 transaksi dengan waktu tunggu pembeli mencapai 11,45 menit. Sedangkan menggunakan aplikasi desktop dengan nomor faktur sebagai antrian proses hanya memperoleh waktu tunggu sebesar 5 menit.



## BAB VI PENUTUP

### 6.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi dan pengujian yang dilakukan, maka diambil kesimpulan sebagai berikut :

1. Aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses dapat melakukan isi pulsa, tambah transaksi, hapus transaksi, menampilkan nomor faktur pembelian, serta dapat mengirimkan *SMS* notifikasi apabila transaksi pulsa yang dilakukan mengalami pending atau gagal.
2. Aplikasi kasir untuk administrator mampu mengecek nomor faktur, check data transaksi berdasarkan status transaksi, serta update ketersediaan voucher maupun harga jual dan harga beli pada menu setting pada aplikasi.
3. Aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses dapat menyelesaikan permasalahan yang terjadi pada sistem lama dengan memanfaatkan nomor faktur sebagai antrian proses dan pembeli sebagai aktor utama dalam proses transaksi pulsa.
4. Penggunaan aplikasi desktop sistem penjualan pulsa dengan nomor faktur sebagai antrian proses lebih efisien karena waktu tunggu dan proses transaksi dari proses penjualan pulsa lebih cepat.
5. *SMS* pending yang diterima oleh *user*, berfungsi sebagai notifikasi ataupun monitoring oleh pembeli terhadap transaksi yang dilakukan. Waktu tunggu *SMS* bergantung pada berapa lama waktu yang diinputkan pada aplikasi gateway sebagai repon kepada server.
6. Rata-rata keseluruhan pada pengujian UAT pada 45 *user* berdasarkan kegunaan fungsi (*percieve usefulness*) memiliki rata-rata sebesar 3,48 dan kemudahan fungsi (*easy of use*) sebesar 3,0. Dengan rata-rata yang didapat pada pengujian diatas, aplikasi tersebut dapat diterima oleh *user* dari segi kegunaan fungsi maupun kemudahan dalam

menjalankan aplikasi tersebut sesuai dengan acuan yang ada pada tabel kuisisioner.

7. Rata-rata keseluruhan pada pengujian UAT pada 5 administrator berdasarkan kegunaan fungsi (*percieve usefulness*) memiliki rata-rata sebesar 3,44 dan kemudahan fungsi (*easy of use*) sebesar 3,12. Dengan rata-rata yang didapat pada pengujian diatas, aplikasi tersebut dapat diterima oleh *user* dari segi kegunaan fungsi maupun kemudahan dalam menjalankan aplikasi tersebut sesuai dengan acuan yang ada pada tabel kuisisioner.

## 6.2 Saran

Saran yang dapat diberikan untuk pengembangan perangkat lunak ini antara lain :

1. Untuk pengembangan lebih lanjut, disarankan memperbaiki *bug-bug* yang ada terutama faktor *componen* pada koneksi modem. Diharapkan dapat menggunakan semua jenis modem sehingga lebih mempermudah proses koneksi antara modem dan aplikasi.
2. Untuk pengembangan lebih lanjut, disarankan aplikasi desktop penjualan pulsa dengan nomor faktur sebagai antrian proses bisa dijalankan secara sistematis dari dalam sistem. Untuk pemilihan operator, pilihan voucher dan nominal bisa otomatis ditampilkan dari dalam sistem apabila terjadi perubahan.
3. Untuk pengembangan lebih lanjut, disarankan untuk antrian proses pada aplikasi kasir disarankan administrator tidak perlu lagi menginputkan nomor faktur tetapi antrian proses dari nomor faktur yang telah didapat pada aplikasi utama tertampil pada menu antrian transaksi.
4. Untuk pemakaian pada counter untuk skala besar dengan jumlah antrian yang banyak, disarankan menggunakan printer kasir untuk proses cetak nomor faktur.

## DAFTAR PUSTAKA

- [AFR-08] Alamsjah, Firdaus; Ramadani, Dax. 2008. *"Using Six Sigma to Improve Business"*.
- [ALT-10] Alvin Leo, Tjatur Kandaga. 2010. Aplikasi Server Pulsa Elektrik dengan *Short Messaging Service (SMS) Gateway dan Sistem Multi Level Marketing (MLM)*. Universitas Kristen Maranatha.
- [AR-10] Aulia, Ratih.2010. *Pembuatan Aplikasi Sistem Penjualan Pulsa Pada Toko Miyo Cell Menggunakan Netbeans 6.0.1 dan MySQL*. Universitas Gunadarma.
- [FA-08] Fikri, Azkal. 2008. *Aplikasi SMS gateway untuk layanan informasi registrasi administrasi mahasiswa, program studi ilmu komputer, Universitas Pendidikan Indonesia"*.
- [MS-06] Madcoms.2006. *Seri Panduan Pemograman : Pemograman Boland Delphi 7*. Andi publisher
- [PRE-01] Pressman, R. 2001. *Software Engineering : A Practitioner's Approach, Fifth Edition*. McGraw Hill.
- [SGH-07] Segar H and Grover, V (2007) Re- examining perceived ease of use measurements and perceived usefulness. *Decision sciences*.
- [SOM-03] Sommerville, Ian. 2003. *Software Engineering, Six Edition. Boston, Addison-Wesley*.
- [SOM-11] Sommerville, Ian. 2011. *Software Engineering, Ninth Edition. Boston, Addison-Wesley*.
- [TFK-08] Taupan Aryo Dwitomo, Firmansyah Apriyadhi, dan Kristian Nalindra Parlinggoman. 2008. *Perancangan Aplikasi SMS Pencatatan KWH Meter dan Penghitungan Tagihan Untuk Pelanggan PT. PLN Lampung (PERSERO)*. Jakarta, Universitas Bina Nusantara.
- [WIY-11] Wiharto, Yudi. 2011. *Sistem Informasi Akademik Berbasis SMS Gateway*. Palembang, Politeknik PalComTech Palembang.