

**RANCANG BANGUN APLIKASI ACTIVATION SIMCARD  
MENGUNAKAN MODEM POOL SEBAGAI ALAT AKTIVASI**

**SKRIPSI**

Konsentrasi Rekayasa Perangkat Lunak

Untuk memenuhi sebagian persyaratan untuk mencapai gelar Sarjana Komputer



Disusun Oleh:

**YAHYA EFENDI**

**NIM. 0910681013**

**PROGRAM STUDI TEKNIK INFORMATIKA  
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER**

**UNIVERSITAS BRAWIJAYA**

**MALANG**

**2013**

## LEMBAR PERSETUJUAN

**RANCANG BANGUN APLIKASI ACTIVATION SIMCARD  
MENGUNAKAN MODEM POOL SEBAGAI ALAT AKTIVASI**

### SKRIPSI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer

UNIVERSITAS BRAWIJAYA



Disusun Oleh:

**YAHYA EFENDI**

**NIM. 0910681013**

Menyetujui :

**Pembimbing I**

**Pembimbing II**

**Denny Sagita Rusdianto, S.Kom., M.Kom**  
**NIK. 851124 06 1 1 0250**

**Fajar Pradana, S.ST., M.Eng**  
**NIK. 850905 16 1 1 0371**

**LEMBAR PENGESAHAN**

**RANCANG BANGUN APLIKASI ACTIVATION SIMCARD  
MENGUNAKAN MODEM POOL SEBAGAI ALAT AKTIVASI**

Disusun oleh:

**Yahya Efendi**

**NIM. 0910681013**

Skripsi ini telah diuji dan dinyatakan lulus pada tanggal 22 Juli 2013

Dosen Penguji I

Dosen Penguji II

**Dr. Eng Herman Tolle, ST., MT.**  
**NIP. 19740823 200012 1 001**

**Aswin Suharsono, ST., MT.**  
**NIK. 840919 06 1 1 0251**

Dosen Penguji III

**Gembong Edhi Setyawan, ST., MT.**  
**NIK. 850920 16 1 1 0373**

Mengetahui,

Ketua Program Studi Teknik Informatika

**Drs. Marji, M.T.**  
**NIP. 19670801 199203 1 001**



## PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata didalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku. (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70)

Malang, Juli 2013

Mahasiswa,

Yahya Efendi

0910681013

## KATA PENGANTAR

*Alhamdulillah rabbil 'alamin.* Puji syukur penulis panjatkan kehadirat Allah SWT, karena atas segala rahmat dan limpahan hidayah-Nya, penulis dapat menyelesaikan tugas akhir yang berjudul “**RANCANG BANGUN APLIKASI ACTIVATION SIMCARD MENGGUNAKAN MODEM POOL SEBAGAI ALAT AKTIVASI**”.

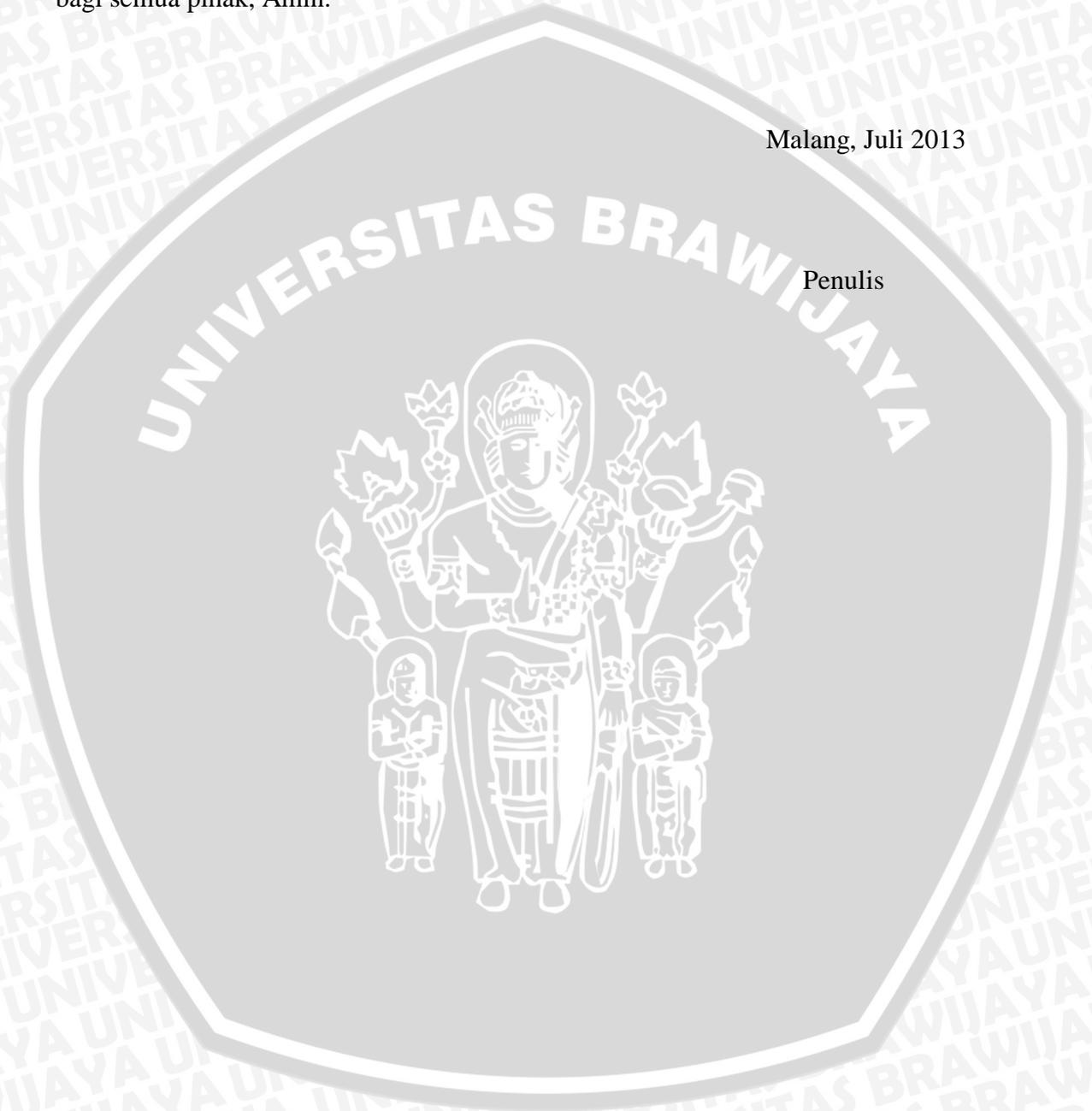
Dalam pelaksanaan dan penulisan tugas akhir ini penulis mendapatkan banyak bantuan dari berbagai pihak baik secara moril maupun materiil. Dalam kesempatan ini penulis ingin mengucapkan terimakasih yang sebesar – besarnya kepada :

- Allah SWT atas rahmat dan hidayah yang telah diberikan.
- Rasulullah Muhammad SAW, semoga shalawat serta salam selalu tercurah kepada beliau.
- Kedua orang tua Ridwan Efendi dan Suhartini, atas segala doa, dorongan, kasih sayang, perhatian, nasehat, dan semangat sehingga penulis dapat menyelesaikan skripsi ini.
- Denny Sagita Rusdianto, S.Kom., M.Kom selaku pembimbing pertama, yang telah memberikan masukan, dorongan, nasehat, ide, saran, serta semangat kepada penulis sehingga penulis dapat menyelesaikan skripsi ini.
- Fajar Pradana, S.ST., M.Eng selaku pembimbing kedua, yang telah memberikan masukan, dorongan, nasehat, ide, saran, serta semangat kepada penulis sehingga penulis dapat menyelesaikan skripsi ini.
- Mas Brilliant Muhammad yang telah berbagi ilmu dan selalu memberi bimbingan dalam menyelesaikan skripsi ini.
- Sahabat sekaligus saudara Achmad Yudi A, Ricky Merdiansyah, Nursetya Ardhi A, dan Rendra Laban G, atas kebersamaan selama ini dan selalu memberikan dorongan semangat serta doa sehingga penulis dapat menyelesaikan skripsi ini.
- Saudara-saudara dan teman-teman atas dorongan semangat dan doa yang tidak dapat penulis sebut satu persatu.

Penulis menyadari bahwa dalam penyusunan tugas akhir ini masih banyak kekurangan baik format penulisan maupun isinya. Oleh karena itu, saran dan kritik membangun dari para pembaca senantiasa penulis harapkan guna perbaikan bagi tugas akhir selanjutnya. Semoga tugas akhir ini dapat memberikan manfaat bagi semua pihak, Amin.

Malang, Juli 2013

Penulis



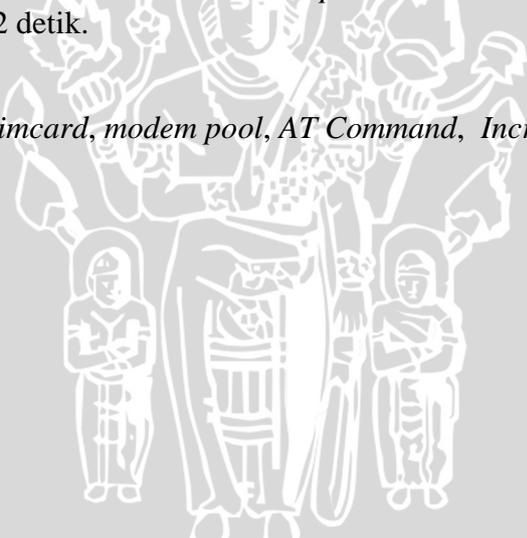
## ABSTRAK

**Yahya Efendi.** 2013. Rancang Bangun Activation Simcard Menggunakan Modem Pool Sebagai Alat Aktivasi. Skripsi Program Studi Teknik Informatika, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya.

Dosen Pembimbing : Denny Sagita Rusdianto, S.Kom., M.Kom dan Fajar Pradana, S.ST., M.Eng

Provider selalu bersaing untuk mendapatkan konsumen dengan menawarkan promo yang murah. Dan provider selalu memiliki *event* khusus yang diberikan ke agen yaitu aktivasi simcard dengan reward yang telah ditentukan. Aktivasi *simcard* dalam jumlah banyak dan pendataan secara manual akan memakan waktu yang lama dan riskan akan *human error*. Pada penelitian ini bertujuan untuk memanfaatkan *modem pool* sebagai alat untuk mempersingkat waktu dalam melakukan aktivasi *simcard* dan menggunakan *AT Command* dalam prosesnya. Metode yang digunakan memanfaatkan proses pengembangan perangkat lunak *incremental model*. Berdasarkan hasil pengujian yang dilakukan, menunjukkan bahwa modem pool dapat dimanfaatkan sebagai alat aktivasi karena semua kebutuhan fungsional dapat terpenuhi. Dan hasil pengujian untuk aktivasi 4 *simcard*, 8 *simcard* dan 16 *simcard* memiliki *persentase* keberhasilan 90,625% dan waktu tunggu 77,52 detik.

**Kata Kunci:** aktivasi *simcard*, *modem pool*, *AT Command*, *Incremental Model*.



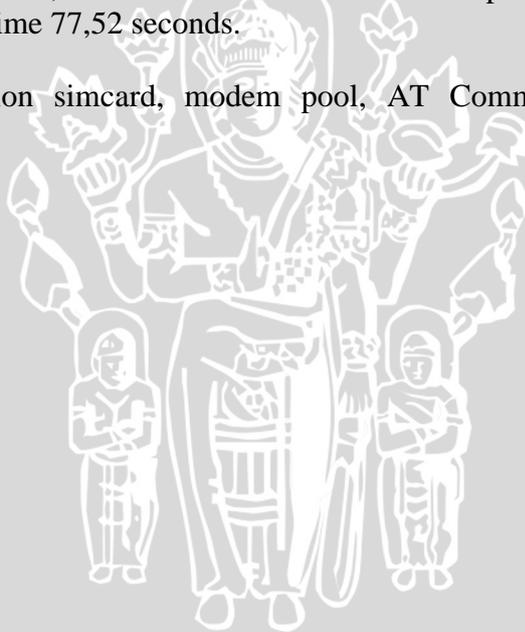
## ABSTRACT

**Yahya Efendi.** 2013. Rancang Bangun Activation Simcard Menggunakan Modem Pool Sebagai Alat Aktivasi. Skripsi Program Studi Teknik Informatika, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya.

Dosen Pembimbing : Denny Sagita Rusdianto, S.Kom., M.Kom dan Fajar Pradana, S.ST., M.Eng

Provider always compete to gain customers by offering cheap promo. And provider always have a special event given to an agent that is the activation simcard with rewards as specified. Activation simcard in great quantities and manual data collection takes much time and susceptible by human error. This research aims to make use of modem pool as a tool to shorten the time to activate the simcard and use the AT Command in the process. That method is used software development process of incremental model. Based on the results of testing carried out, indicating that the modem pool can be used as a means of activation as all functional requirements can be met. And the results of testing for the activation of 4 simcard, 8 simcard and 16 simcard has a percentage of success 90,625% and waiting time 77,52 seconds.

**Kata Kunci:** activation simcard, modem pool, AT Command, Incremental Model.



**DAFTAR ISI**

LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN .....	iii
KATA PENGANTAR .....	vi
ABSTRAK.....	viii
ABSTRACT.....	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR .....	xii
DAFTAR TABEL.....	xiv
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah .....	2
1.4. Tujuan.....	3
1.5. Manfaat.....	3
1.6. Sistematika Pembahasan.....	3
<b>BAB II DASAR TEORI.....</b>	<b>5</b>
2.1 <i>Short Message Service</i> .....	5
2.2 <i>SMS Gateway</i> .....	6
2.3 <i>AT Command</i> .....	6
2.4 Peraturan Menteri Komunikasi Dan Informatika .....	7
2.5 Aktivasi <i>Simcard</i> .....	7
2.6 Rekayasa Perangkat Lunak.....	8
2.7 Pengujian Perangkat Lunak .....	10
2.7.1 <i>Black-Box Testing</i> .....	10
2.7.2 Pengujian Validasi .....	11
2.7.3 Pengujian Performa.....	11
<b>BAB III METODOLOGI PENELITIAN.....</b>	<b>12</b>
3.1 Studi Literatur.....	12
3.2 Metode Pengambilan Data.....	12
3.3 Analisis Kebutuhan.....	12

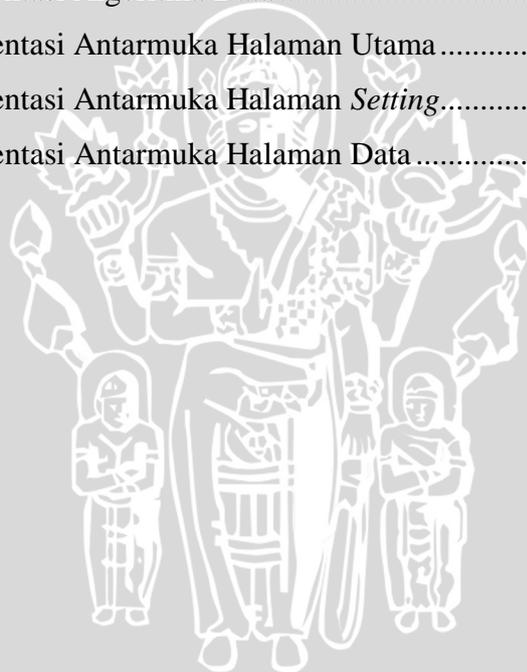


3.4	Perancangan dan Implementasi .....	13
3.5	Pengujian dan Analisis .....	14
3.6	Pengambilan Kesimpulan .....	15
<b>BAB IV PERANCANGAN IMPLEMENTASI .....</b>		<b>16</b>
4.1	Perancangan.....	16
4.1.1	Analisis Kebutuhan.....	17
4.1.2	Perancangan Perangkat Lunak.....	34
4.2	Implementasi .....	45
4.2.1	Spesifikasi Sistem.....	46
4.2.2	Batasan Implementasi.....	47
4.2.3	Implementasi Basis Data .....	48
4.2.4	Implementasi Algoritma .....	48
4.2.5	Implementasi Antarmuka.....	55
<b>BAB V PENGUJIAN DAN ANALISIS.....</b>		<b>58</b>
5.1	Pengujian .....	58
5.1.1	Pengujian Validasi.....	58
5.1.2	Pengujian Performa.....	64
5.2	Analisis .....	68
5.2.1	Analisis Hasil Pengujian Validasi.....	68
5.2.2	Analisis Hasil Pengujian Performa.....	68
<b>BAB VI PENUTUP .....</b>		<b>70</b>
6.1	Kesimpulan.....	70
6.2	Saran .....	70
<b>DAFTAR PUSTAKA .....</b>		<b>71</b>

## DAFTAR GAMBAR

Gambar 2.1 Diagram Alir Aktivasi <i>Simcard</i> .....	8
Gambar 2.2 <i>Waterfall Model</i> .....	10
Gambar 3.1 Diagram Alir Implementasi Aplikasi <i>Activation Simcard</i> .....	14
Gambar 3.2 Diagram Alir Pengujian dan Analisis Aplikasi <i>Activation Simcard</i> .	15
Gambar 4.1 Diagram Blok Perancangan.....	16
Gambar 4.2 Gambar Alat Aktivasi Manual .....	17
Gambar 4.3 Sistem Aplikasi <i>Activation Simcard</i> .....	18
Gambar 4.4 Diagram <i>Use Case</i> Agen .....	23
Gambar 4.5 Diagram Aktifitas Melakukan Setting.....	24
Gambar 4.6 Diagram Aktifitas Melakukan Connect.....	25
Gambar 4.7 Diagram Aktifitas Melakukan Registrasi .....	27
Gambar 4.8 Diagram Aktifitas Melakukan SMS.....	28
Gambar 4.9 Diagram Aktifitas Melakukan Telepon.....	29
Gambar 4.10 Diagram Aktifitas Melakukan <i>Transfer</i> Pulsa .....	30
Gambar 4.11 Diagram Aktifitas Melakukan Pengecekan.....	31
Gambar 4.12 Diagram Aktifitas Mengecek <i>Inbox</i> .....	32
Gambar 4.13 Diagram Aktifitas Melakukan Penyimpanan Data.....	33
Gambar 4.14 Diagram Aktifitas Melihat Data.....	35
Gambar 4.15 Diagram <i>Entity Relationship</i> Perangkat Lunak <i>Activation Simcard</i> dengan Menggunakan <i>Modem Pool</i> Sebagai Alat Aktivasi.....	36
Gambar 4.16 Perancangan Algoritma Halaman <i>Setting</i> .....	37
Gambar 4.17 Perancangan Algoritma Connect .....	38
Gambar 4.18 Perancangan Algoritma Registrasi .....	39
Gambar 4.19 Perancangan Algoritma SMS .....	39
Gambar 4.20 Perancangan Algoritma Telepon.....	40
Gambar 4.21 Perancangan Algoritma <i>Transfer</i> Pulsa.....	40
Gambar 4.22 Perancangan Algoritma Pengecekan.....	41
Gambar 4.23 Perancangan Algoritma Mengecek <i>Inbox</i> .....	41
Gambar 4.24 Perancangan Algoritma <i>Save</i> .....	42

Gambar 4.25 Perancangan Algoritma Data .....	42
Gambar 4.26 Tampilan Antarmuka Halaman Utama .....	43
Gambar 4.27 Tampilan Antarmuka <i>Groupbox</i> .....	44
Gambar 4.28 Tampilan Antarmuka Halaman Setting.....	45
Gambar 4.29 Tampilan Antarmuka Halaman Data.....	46
Gambar 4.30 Diagram Blok Implementasi .....	47
Gambar 4.31 Diagram ER Konseptual dari Perangkat Lunak <i>Activation Simcard</i> dengan Menggunakan <i>Modem Pool</i> Sebagai Alat Aktivasi.....	49
Gambar 4.32 Implementasi Algoritma Connect .....	50
Gambar 4.33 Implementasi Algoritma <i>Save</i> .....	52
Gambar 4.34 Implementasi Algoritma <i>Setting</i> .....	54
Gambar 4.35 Implementasi Algoritma Data .....	56
Gambar 4.36 Implementasi Antarmuka Halaman Utama .....	57
Gambar 4.37 Implementasi Antarmuka Halaman <i>Setting</i> .....	57
Gambar 4.38 Implementasi Antarmuka Halaman Data.....	58



## DAFTAR TABEL

Tabel 2.1 Tabel <i>At Command</i> .....	6
Tabel 4.1 Identifikasi aktor .....	20
Tabel 4.2 Spesifikasi Kebutuhan Fungsional Agen .....	21
Tabel 4.3 <i>Use case</i> Melakukan Setting .....	23
Tabel 4.4 <i>Use Case</i> Melakukan Connect .....	25
Tabel 4.5 <i>Use Case</i> Melakukan Registrasi .....	26
Tabel 4.6 Use Case Melakukan SMS.....	27
Tabel 4.7 Use Case Melakukan Telepon.....	28
Tabel 4.8 Use Case Melakukan <i>Transfer</i> Pulsa .....	29
Tabel 4.9 Use Case Melakukan Pengecekan.....	30
Tabel 4.10 Use Case Mengecek <i>Inbox</i> .....	32
Tabel 4.11 Use Case Melakukan Penyimpanan Data .....	33
Tabel 4.12 Use Case Melihat Data.....	34
Tabel 4.13 Struktur tabel <i>dataact</i> .....	36
Tabel 4.14 Struktur tabel <i>sett_app</i> .....	36
Tabel 4.15 Spesifikasi Perangkat Keras Komputer.....	47
Tabel 4.16 Spesifikasi Perangkat Lunak Komputer.....	48
Tabel 5.1 Kasus Uji Melakukan <i>Setting</i> .....	60
Tabel 5.2 Kasus Uji Melakukan Connect .....	60
Tabel 5.3 Kasus Uji Melakukan Registrasi.....	61
Tabel 5.4 Kasus Uji Melakukan SMS.....	61
Tabel 5.5 Kasus Uji Melakukan Telepon.....	62
Tabel 5.6 Kasus Uji Melakukan <i>Transfer</i> Pulsa .....	62
Tabel 5.7 Kasus Uji Melakukan Pengecekan.....	63
Tabel 5.8 Kasus Uji Melakukan Mengecek <i>Inbox</i> .....	64
Tabel 5.9 Kasus Uji Melakukan Melakukan Penyimpanan Data .....	64
Tabel 5.10 Kasus Uji Melihat Data.....	65
Tabel 5.11 Hasil Pengujian Performa Satu <i>Simcard</i> .....	66
Tabel 5.12 Hasil Pengujian Performa 4 <i>Simcard</i> .....	67

Tabel 5.13 Hasil Pengujian Performa 8 *Simcard* ..... 67

Tabel 5.14 Hasil Pengujian Performa 16 *Simcard* ..... 68



## BAB I PENDAHULUAN

### 1.1. Latar Belakang

Teknologi sekarang terus berkembang dan salah satunya adalah teknologi komunikasi. Komunikasi sangat diperlukan karena semua informasi dapat disalurkan dengan cepat apabila ada komunikasi. Di beberapa tahun terakhir ini, *SMS (Short Message Service)* telah membuat dampak besar pada cara kita berkomunikasi [MSK-09]. SMS dapat menyebarkan informasi dengan cepat dan biaya yang tergolong murah, sehingga komunikasi dapat dilakukan dengan mudah. Dengan media komunikasi konvergen, SMS kini dapat diakses dalam banyak hal sebagai alat bisnis [MSK-09].

Selain itu, produk-produk telepon seluler pun terus menawarkan teknologi yang canggih dengan harga yang murah. Hal itu akan berdampak terus meningkatnya user atau pengguna telepon seluler. Pengguna telepon seluler sebagai konsumen akan memilih *provider* yang menawarkan biaya yang murah dalam melakukan komunikasi. Hal ini yang membuat *provider* terus bersaing dalam menawarkan biaya yang murah melalui promo yang diberikan.

*Provider* pulsa memberikan *event* khusus kepada agen-agen pulsa yang memiliki ID dengan *provider* tersebut. Agen pulsa ini memiliki Nomor ID pada masing-masing *provider* atas nama satu orang saja. Setiap *event* yang diberikan ke agen-agen pulsa mengharuskan agen-agen pulsa membeli dan mengaktifkan kartu-kartu perdana dengan jumlah tertentu untuk mendapat keuntungan. Jumlah yang harus diaktivasi pada *event* tersebut selalu berjumlah banyak dan memiliki ketentuan tertentu. Jumlah kartu dalam event tersebut adalah aktivasi 20000 *simcard* dalam waktu 1 bulan.

Pengaktifan dalam jumlah banyak akan sangat sulit apabila dilakukan secara manual, karena pengaktifan secara manual akan memakan waktu yang sangat lama. Pendataan yang dilakukan dalam jumlah banyak juga akan memakan waktu yang lama dan sangat riskan karena *human error*. Sehingga sangat memungkinkan pendataan kurang akurat karena bisa terlewat dalam pendataan.

Pengaktifan secara manual yang memakan waktu lama dapat diatasi dengan menggunakan *modem pool* yang dapat melakukan aktivasi dengan menggunakan

SMS Gateway. *Modem pool* memiliki fungsi hampir sama seperti *modem* biasa, yang membedakan adalah *modem* ini memiliki slot yang banyak untuk *simcard*. Sehingga penggunaan *modem pool* akan menjadi solusi dimana nanti akan dilakukan pengujian lama waktu pengaktifan *simcard* dalam jumlah banyak dengan menggunakan *modem pool*.

Berdasarkan masalah diatas maka dibuatlah aplikasi *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi. *Modem pool* memiliki jumlah *port* yang banyak sehingga pengaktifan pun dapat dilakukan dengan jumlah banyak dalam satu kali proses. Selain itu pendataan juga dapat dilakukan setelah pengaktifan dengan langsung mengambil data dan disimpan pada *database*. Sehingga diharapkan dengan aplikasi ini dapat melakukan pengaktifan dalam waktu yang lebih cepat dan melakukan pendataan lebih akurat karena tidak riskan akan *human error*.

## 1.2. Rumusan Masalah

Berdasarkan paparan latar belakang tersebut, maka rumusan masalah adalah sebagai berikut:

1. Menganalisa kebutuhan agen pulsa untuk merancang aplikasi *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi.
2. Mengembangkan dan mengimplementasikan aplikasi *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi.
3. Menguji aplikasi *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi.

## 1.3. Batasan Masalah

Batasan masalah dalam penelitian ini adalah :

1. *Provider* yang digunakan untuk di aktivasi adalah indosat.
2. *SMS Gateway* pada aplikasi ini digunakan untuk registrasi.
3. *Basis data* yang digunakan adalah *basis data MySQL*.
4. Bahasa pemrograman yang digunakan pada aplikasi ini adalah bahasa pemrograman Delphi.

#### 1.4. Tujuan

Tujuan dari penelitian ini adalah:

1. Mengetahui kebutuhan agen pulsa untuk merancang aplikasi *Activation Simcard* dengan menggunakan *modem pool*.
2. Dapat mengembangkan dan mengimplementasikan aplikasi *Activation Simcard* dengan menggunakan *modem pool*.
3. Menguji aplikasi *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi sesuai dengan kebutuhan.

#### 1.5. Manfaat

Manfaat dari penelitian ini adalah:

1. Bagi Penulis
  1. Mendapatkan pengetahuan tentang pembuatan aplikasi dengan memanfaatkan *modem pool*.
  2. Mendapatkan pengetahuan tentang prosedur aktivasi *simcard*.
2. Bagi Pengguna
  1. Aplikasi *Activation Simcard* dapat digunakan sebagai aplikasi yang dapat mengaktifkan *simcard* langsung beberapa *simcard* dalam satu kali aktivasi
  2. Aplikasi *Activation Simcard* digunakan sebagai aplikasi yang dapat mengaktifkan *simcard* lebih cepat dan membuat pendataan yang akurat.

#### 1.6. Sistematika Pembahasan

Untuk mencapai tujuan yang diharapkan, maka sistematika penulisan yang disusun dalam tugas akhir ini adalah sebagai berikut:

### 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 perancangan aplikasi *Activation Simcard* dengan menggunakan *modem pool*.

## **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 Implementasi**

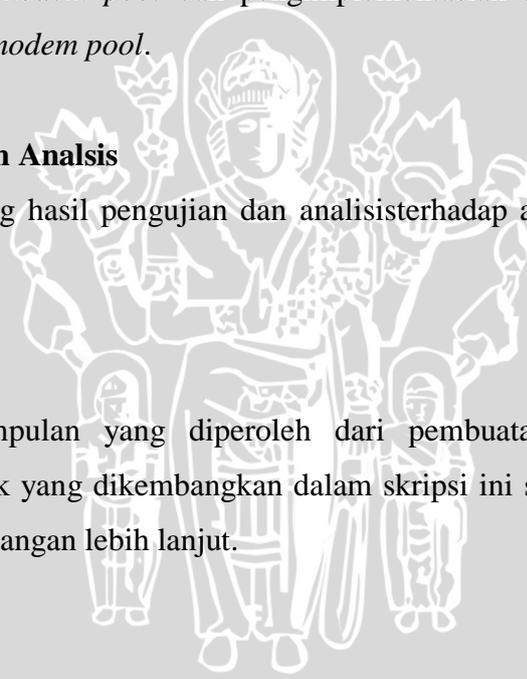
Membahas tentang perancangan dari aplikasi *Activation Simcard* dengan menggunakan *modem pool* dan pengimplementasian aplikasi *Activation Simcard* pada *modem pool*.

## **BAB V Pengujian dan Analisis**

Memuat tentang hasil pengujian dan analisis terhadap aplikasi 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

### DASAR TEORI

Bab ini membahas teori yang diperlukan untuk menyusun penelitian yang diusulkan. Dasar teori yang digunakan adalah *Short Message Service (SMS)*, *SMS Gateway*, *AT Command* dan rekayasa perangkat lunak.

#### 2.1 *Short Message Service*

*Short Message Service (SMS)* adalah kemampuan untuk mengirim dan menerima pesan dalam bentuk teks dari dan kepada ponsel. Teks tersebut bisa terdiri dari kata-kata atau nomor atau kombinasi alphanumeric. SMS diciptakan sebagai standar pesan oleh *ETSI (European Telecommunication Standards Institute)*, yang juga membuat standar GSM yang diimplementasikan oleh semua operator GSM. Sejarah SMS muncul pada Desember 1992. Pesan itu dikirim dari sebuah PC ke sebuah telepon seluler dalam jaringan GSM milik operator seluler Vodafone di Inggris [TFK-08]. Setiap pesan singkat dapat menjadi tidak lebih dari 160 karakter, sedangkan karakter ini bisa berupa teks (alfanumerik) atau biner pesan pendek Non-Teks [KAV-10].

*Short Message Service* atau biasa disingkat SMS merupakan sebuah layanan yang banyak diaplikasikan pada sistem komunikasi tanpa kabel (*wireless*), memungkinkan dilakukannya pengiriman pesan dalam bentuk alphanumeric antara terminal pelanggan atau antar terminal pelanggan dengan sistem eksternal, seperti *e-mail*, *paging*, *voice mail*, dan lain-lain. Aplikasi SMS merupakan aplikasi yang paling banyak peminat dan penggunaannya. Hal ini dapat dibuktikan dengan munculnya berbagai jenis aplikasi yang memanfaatkan fasilitas SMS.

Teknologi SMS memiliki beberapa keunggulan, yaitu harganya murah, merupakan "*Delivered Oriented Service*", artinya pesan akan selalu diusahakan untuk dikirimkan ke tujuan. Jika suatu saat nomor tujuan sedang tidak aktif atau di luar jaringan, maka pesan akan disimpan di *SMSC (SMS Center)* server dan akan dikirimkan segera setelah nomor tujuan aktif kembali. Pesan juga akan terkirim ke tujuan walaupun nomor tujuan sedang melakukan pembicaraan (sibuk).

Cara kerja SMS : Saat kita menerima pesan SMS/MMS dari *handphone (mobile originated)*, pesan tersebut tidak langsung dikirimkan ke *handphone*

tujuan (*mobile terminated*), akan tetapi dikirim terlebih dahulu ke *SMS Center* (SMSC) yang biasanya berada di kantor operator telepon, baru kemudian pesan tersebut diteruskan ke *handphone* tujuan. Dengan adanya SMSC, kita dapat mengetahui status dari pesan SMS yang telah dikirim, apakah telah sampai atau gagal.

## 2.2 SMS Gateway

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 disimpan pada *database*.

## 2.3 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*) [TFK-08]. Umumnya *AT Command* digunakan pada *Hyper Terminal*. Dibawah ini merupakan beberapa contoh *AT Command* :

**Tabel 2.1 Tabel AT Command**  
Sumber : [WIY-11]

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 SMS yang baru masuk secara otomatis
AT+CMGL	Membuka daftar SMS yang ada pada SIM Card
AT+CMGS	Mengirim Pesan SMS
AT+CMGR	Membaca Pesan SMS
AT+CMGD	Menghapus Pesan SMS

#### 2.4 Peraturan Menteri Komunikasi Dan Informatika

Menteri komunikasi dan informatika telah menetapkan peraturan pada tahun 2005 tentang registrasi terhadap pelanggan jasa telekomunikasi. Pada peraturan tersebut telah disebutkan bahwa pelanggan mempunyai hak menggunakan jasa telekomunikasi setelah memberikan identitasnya secara benar kepada penyelenggara telekomunikasi yang ada pada pasal 2 [DJA-05].

Selain itu disebutkan pada pasal 4 ayat 1 bahwa penyelenggara jasa telekomunikasi wajib menerapkan registrasi untuk setiap pelanggan jasa telekomunikasi pra bayar dan memiliki identitas pelanggan dimaksud. Dan ayat 2 bagian b menyebutkan identitas yang terdapat pada Kartu Tanda Penduduk/Surat Izin Mengemudi/Pasport/Kartu Pelajar, yaitu nomor, nama, tempat/tanggal lahir dan alamat [DJA-05]. Dari peraturan yang telah ditetapkan pelanggan dapat memiliki banyak nomor dari penyelenggara jasa telekomunikasi dengan ketentuan melakukan registrasi dengan data yang telah ditentukan oleh penyelenggara jasa telekomunikasi.

#### 2.5 Aktivasi Simcard

Aktivasi simcard dilakukan tidak hanya melakukan registrasi ke *provider*, tetapi ada beberapa tahap selain registrasi yang harus dilakukan untuk memenuhi syarat dalam mengikuti sebuah *event*. Setiap *provider* selalu memiliki prosedur yang telah mereka tentukan dalam melakukan aktivasi *simcard*. Dalam skripsi dibuat urutan prosedur sesuai yang ditentukan oleh indosat.

Prosedur dari indosat dalam melakukan aktivasi untuk memenuhi syarat dalam mengikuti event memiliki beberapa prosedur yaitu *registrasi*, *sms claim*, dan *call*. Selain *prosedur registrasi*, *sms claim*, dan *call* ada *prosedur* tambahan yang harus dilakukan oleh agen pulsa yaitu melakukan pendataan dan menyimpan

data *simcard* yang telah di aktivasi. Proses aktivasi *simcard* secara keseluruhan dapat dilihat dalam Gambar 2.1.



**Gambar 2.1 Diagram Alir Proses Aktivasi *Simcard***

## 2.6 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-11]. 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.

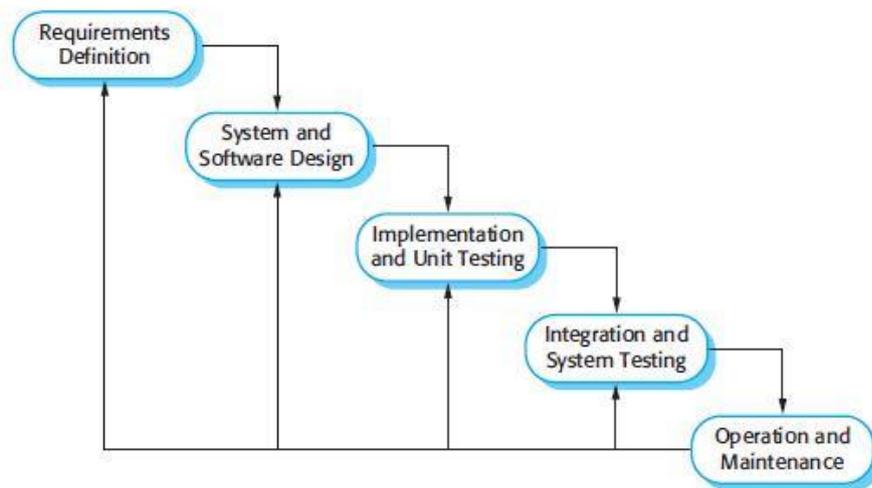
2. ‘semua aspek produksi perangkat lunak’

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-11].

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 2.2 Waterfal Model  
Sumber : [SOM-11]

## 2.7 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*.

### 2.7.1 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.

### 2.7.2 Pengujian Validasi

Pada kulminasi pengujian terintegrasi, perangkat lunak secara lengkap dirakit sebagai suatu paket; kesalahan *interfacing* telah diungkap dan dikoreksi, dan seri akhir dari pengujian perangkat lunak, yaitu pengujian validasi dapat dimulai. Validasi dapat ditentukan dengan berbagai cara, tetapi definisi yang sederhana adalah bahwa validasi berhasil bila perangkat lunak berfungsi dengan cara yang dapat diharapkan secara bertanggung jawab oleh pelanggan. Validasi perangkat lunak dicapai melalui sederetan pengujian *black-box* yang memperlihatkan konformitas dengan persyaratan. Rencana pengujian menguraikan kelas-kelas pengujian yang akan dilakukan, dan prosedur pengujian menentukan *test case* spesifik yang akan digunakan untuk mengungkap kesalahan dalam konformitas dengan persyaratan. Baik rencana dan prosedur didesain untuk memastikan apakah semua persyaratan fungsional dipenuhi, semua persyaratan kinerja dicapai, dokumentasi betul dan direkayasa oleh manusia, dan persyaratan lainnya dipenuhi (transportabilitas, kompatibilitas, pembetulan kesalahan, maintainabilitas) [PRE-01].

### 2.7.3 Pengujian Performa

Setelah semua langkah pengujian validasi 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 aktivasi. 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

Pada bab ini akan dibahas metode-metode yang akan digunakan dalam perancangan sistem yaitu studi literatur, metode pengambilan data, analisis kebutuhan, perancangan sistem, implementasi, pengujian dan analisis, dan pengambilan kesimpulan.

#### 3.1 Studi Literatur

Studi literatur menjelaskan dasar teori yang digunakan sebagai penunjang dan pendukung penulisan skripsi. Teori penunjang dan pendukung skripsi ini meliputi *SMS Gateway*, *AT Command*, aktivasi *Simcard*, *incremental model*, pemrograman Delphi dan pengujian perangkat lunak. Sumber atau referensi yang digunakan antara lain buku, jurnal, laporan penelitian, dan bantuan mesin pencari (*search engine*) internet.

#### 3.2 Metode Pengambilan Data

Metode ini digunakan untuk mendapatkan data sebagai acuan untuk pengembangan pengembangan perangkat lunak. Data yang dimaksud adalah data untuk registrasi dan data *simcard* yang diaktifkan yang digunakan untuk implementasi, serta data jumlah *simcard* yang harus diaktifkan dan waktu yang diberikan yang digunakan untuk menghitung tingkat kesuksesan perangkat lunak *Activation Simcard*. Data-data tersebut digunakan untuk menghitung tingkat keberhasilan dari perangkat lunak.

#### 3.3 Analisis Kebutuhan

Analisis kebutuhan perangkat lunak meliputi analisis spesifikasi perangkat lunak. Metode analisis menggunakan bahasa pemodelan UML (*Unified Modeling Language*). *Use Case Diagram* digunakan 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 yang nantinya akan disediakan oleh aplikasi ini antara lain adalah :

1. Aplikasi pada komputer ini digunakan untuk kartu perdana Indosat yang menyediakan fasilitas untuk registrasi, *call*, *SMS* , *transfer* pulsa, dan menyimpan data pada *database*.
2. Aplikasi ini memanfaatkan *modem pool* sebagai alat untuk pengaktifan untuk memenuhi semua kebutuhan tersebut.
3. Registrasi dan *SMS* menggunakan *SMS Gateway* untuk *SMS* ke *provider* Indosat.

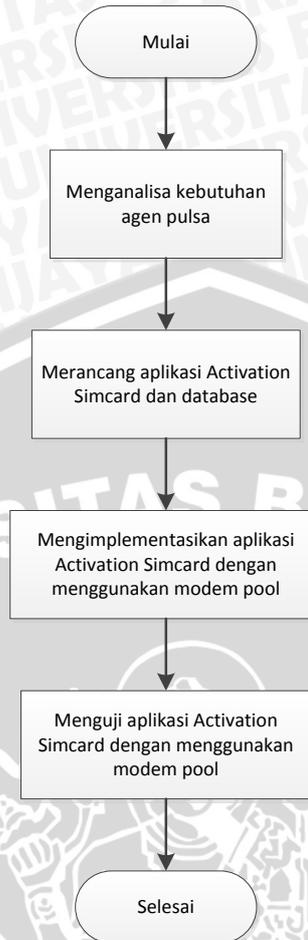
#### 3.4 Perancangan dan Implementasi

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 aplikasi *Activation Simcard* dengan menggunakan *modem pool*. Perancangan sistem berdasarkan *Object Oriented Analysis* dan *Object Oriented Design* yaitu menggunakan pemodelan UML (*Unified Modeling Language*).Basis data yang dimiliki oleh aplikasi hanya satu, yang berada di komputer administrator dimana aplikasi *Activation Simcard* akan digunakan.

Dan tahap implementasi menjelaskan tahapan yang dilakukan untuk merealisasikan aplikasi *Activation Simcard* yang mengacu pada perancangan sistem. Tahap yang dilakukan antara lain:

- Membuat basis data dengan *Mysql* sesuai dengan desain basis data dalam perancangan aplikasi *Activation Simcard*.
- Mendesain aplikasi sesuai dengan perancangan pada sistem menggunakan pemrograman Delphi.
- Implementasi *SMS Gateway* menggunakan *AT Command* dengan menggunakan alat aktivasi *modem pool*.
- Melakukan pengecekan aplikasi *Activation Simcard* telah sesuai dengan kebutuhan.

Diagram alir implementasi aplikasi *Activation Simcard* ini akan ditunjukkan dalam Gambar 3.1.



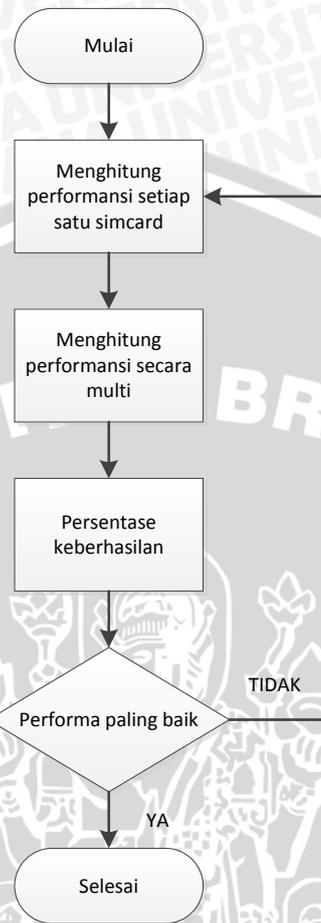
**Gambar 3.1 Diagram Alir Implementasi Aplikasi Activation Simcard**  
Sumber : [Perancangan]

Dalam proses pengembangan perangkat lunak menggunakan *waterfall model* yang pertama dilakukan adalah menganalisa kebutuhan agen pulsa. Setelah kebutuhan agen pulsa benar-benar dipahami maka dilanjutkan proses selanjutnya yaitu dengan merancang aplikasi *Activation Simcard* dan *database*. Proses selanjutnya dapat dilakukan apabila proses sebelumnya telah selesai. Proses selanjutnya adalah mengimplementasikan aplikasi dengan menggunakan *modem pool*. Dan proses yang terakhir adalah menguji aplikasi *Activation Simcard* dengan menggunakan *modem pool*.

### 3.5 Pengujian dan Analisis

Pengujian dari perangkat lunak ini berkaitan dengan pengujian sistem (*black box*). Analisis sistem dilakukan dengan membandingkan data hasil *sample* sebelum dan sesudah sistem aplikasi diimplementasikan sehingga didapatkan

suatu kesimpulan. Diagram alir pengujian dan analisis ditunjukkan dalam Gambar 3.2.



**Gambar 3.2 Diagram Alir Pengujian dan Analisis Aplikasi *Activation Simcard***  
Sumber : [Perancangan]

### 3.6 Pengambilan Kesimpulan

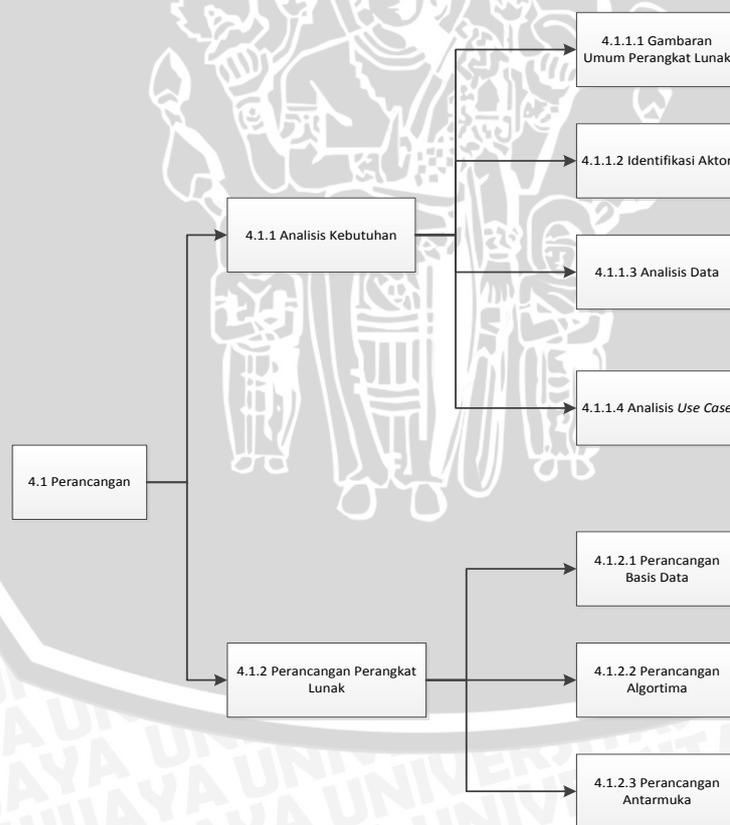
Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi dan pengujian sistem telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan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

## BAB IV PERANCANGAN IMPLEMENTASI

Pada bab ini dibahas mengenai implementasi sistem berdasarkan hasil yang telah diperoleh dari analisis kebutuhan dan proses perancangan perangkat lunak yang dibuat. Pembahasan terdiri dari penjelasan tentang spesifikasi sistem, batasan-batasan dalam implementasi, implementasi algoritma pada program, implementasi antarmuka, dan implementasi metode.

### 4.1 Perancangan

Perancangan aplikasi *Activation Simcard* dengan menggunakan *modem pool* 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 seperti yang digambarkan dalam Gambar 4.1 berikut ini.



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

#### 4.1.1 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 *Activation Simcard*, identifikasi aktor yang terlibat, analisis data yang akan disimpan, penjabaran tentang daftar kebutuhan 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.

##### 4.1.1.1 Gambaran umum perangkat lunak *Activation Simcard* menggunakan *modem pool*

Perangkat lunak ini adalah perangkat lunak yang digunakan untuk membantu mengaktifkan *simcard* dengan menggunakan *modem pool* sebagai alat aktivasi. Sebelumnya pengaktifan dilakukan secara manual yang mengharuskan user memasukkan data secara manual dan pendataan yang dilakukan secara manual. Alat yang digunakan pada waktu pengaktifan manual adalah handphone yang telah dimodifikasi agar memudahkan seperti dalam Gambar 4.2 berikut ini



Gambar 4.2 Gambar Alat Aktivasi Manual  
Sumber : [Perancangan]

Pengaktifan *simcard* secara manual dilakukan menggunakan *handphone* yang telah dimodifikasi tetap mengharuskan user memasukkan satu persatu untuk setiap pengaktifan. *User* juga diharuskan melakukan panggilan dan pendataan secara manual yang akan memakan waktu yang sangat lama. Selain itu juga riskan akan terjadinya *human error*.

Perangkat lunak yang akan dibuat ini akan memanfaatkan *modem pool* sebagai alat aktivasi sehingga aktivasi dapat dilakukan langsung beberapa *simcard*. Pengiriman data registrasi menggunakan *SMS Gateway* dan penyimpanan data tidak dilakukan secara manual tetapi langsung dimasukkan ke *database* yang telah dibuat sebelumnya. Perangkat lunak ini memiliki sistem seperti dalam Gambar 4.4 berikut ini.



Gambar 4.3 Sistem Aplikasi Activation Simcard  
Sumber : [Perancangan]

*User* atau agen yang akan mengaktifkan *simcard* harus mengisi data terlebih dahulu di sistem atau aplikasi yang telah dibuat. Dengan aplikasi ini *user* tidak perlu mengaktifkan satu persatu karena bisa langsung mengaktifkan 16 *simcard* karena menggunakan *modem pool* dengan 16 *port*. Sehingga waktu pengaktifan akan lebih cepat. Perangkat lunak ini memiliki empat fungsi utama yaitu :

a. Fungsi *Connect*

Fungsi *connect* pada perangkat lunak ini berfungsi untuk menjalankan timer utama dan akan menjalankan *timer* pada setiap *groupbox* yang telah dibuat. Proses yang terjadi setelah *timer* berjalan adalah menjalankan *AT Command* yang telah dibuat sebagai berikut :

1.  $AT+CMGF=1$

*AT Command* ini berfungsi untuk mengubah modem menjadi *mode text*

2.  $AT+CPMS="SM"$

*AT Command* ini berfungsi untuk mengatur penyimpanan data akan disimpan di *simcard*

3. AT+CSCS="GSM"

*AT Command* ini berfungsi untuk mengatur karakteristik yang digunakan adalah GSM

4. AT+CREG=1

*AT Command* ini berfungsi untuk registrasi *network*

5. AT+CLCK="FD"

*AT Command* ini berfungsi untuk unlock *simcard* dan yang di unlock adalah *fixed dial*

6. ATD

*AT Command* ini berfungsi untuk telepon

7. ATH

*AT Command* ini berfungsi untuk hang up atau menutup telepon

8. AT+CMGS

*AT Command* ini berfungsi untuk mengirim sms

9. AT+CUSD=1

*AT Command* ini berfungsi untuk melakukan dial ke USSD (*Unstructured Supplementary Service Data*)

b. Fungsi *Save*

Fungsi *save* pada perangkat lunak ini berfungsi untuk menyimpan semua data nomor, masa aktif, waktu aktivasi dan status aktivasi.

c. Fungsi *Setting*

Fungsi *Setting* pada perangkat lunak ini berfungsi untuk mengatur semua data yang diperlukan seperti data registrasi, *mastertransfer* pulsa, nomor untuk *call*, dll.

d. Fungsi *Data*

Fungsi *data* pada perangkat lunak ini berfungsi untuk melihat data yang selesai di aktivasi dan melihat jumlah data yang tersimpan di *database*.

#### 4.1.1.2 Identifikasi Aktor

Tahap ini adalah tahap untuk melakukan identifikasi terhadap aktor yang akan berinteraksi dengan aplikasi *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi. 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
Agen	Agen adalah pengguna yang dapat menggunakan semua fitur yang ada yaitu melakukan registrasi, <i>calling</i> , <i>transfer</i> pulsa, bisa melakukan <i>save</i> untuk menyimpan data pada <i>database</i> yang telah dibuat untuk pendataan dan dapat melihat data yang ada di <i>database</i> .

Sumber : [Perancangan]

#### 4.1.1.3 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 :

- a. Data Aktivasi yang terdiri dari id, MSISDN (*Mobile Subscriber Integrated Services Digital Network*), masa aktif, waktu aktivasi dan status aktivasi. Data ini berfungsi untuk menyimpan data simcard telah berhasil di aktivasi.
- b. Data setting yang terdiri dari id, nama setting, isi setting dan tujuan. Data ini berfungsi untuk menyimpan data pengaturan sehingga tidak mengatur ulang setiap aplikasi di buka kembali.

#### 4.1.1.4 Analisis Use Case

Analisis *use case* dibagi mejadi 2, yaitu spesifikasi kebutuhan fungsional agen dan diagram *use case*. Spesifikasi kebutuhan fungsional agen ditunjukkan dalam Tabel 4.2.

**Tabel 4.2 Spesifikasi Kebutuhan Fungsional Agen**

Nomor SRS	Kebutuhan	Use Case
SRS_001	Perangkat lunak harus mampu menyimpan data untuk <i>setting</i> pada aplikasi.	Melakukan <i>Setting</i>

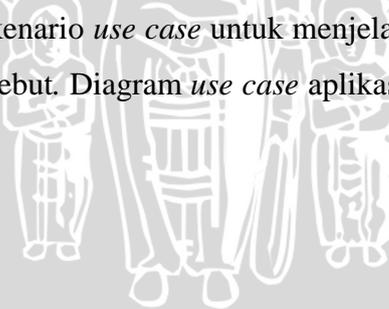
SRS_002	Perangkat lunak harus mampu melakukan semua proses aktivasi untuk semua simcard tanpa harus satu persatu.	Melakukan <i>Connect</i>
SRS_003	Perangkat lunak harus mampu mengirim sms menggunakan <i>modem pool</i> untuk melakukan <i>registrasi</i> .	Melakukan <i>Registrasi</i>
SRS_004	Perangkat lunak harus mampu mengirim sms menggunakan <i>modem pool</i> untuk melakukan sms atau bisa dimanfaatkan untuk claim.	Melakukan SMS
SRS_005	Perangkat lunak harus mampu telepon atau dial up menggunakan <i>modem pool</i> .	Melakukan Telepon
SRS_006	Perangkat lunak harus mampu melakukan <i>transfer</i> pulsa menggunakan <i>modem pool</i> .	Melakukan <i>Transfer</i> Pulsa
SRS_007	Perangkat lunak harus mampu melakukan pengecekan nomor dan masa aktif menggunakan <i>modem pool</i> .	Melakukan pengecekan
SRS_008	Perangkat lunak harus mampu mengecek <i>inbox</i> bila ada sms masuk <i>kesimcard</i> telah mendapatkan laporan registrasi berhasil.	Mengecek <i>Inbox</i>
SRS_009	Perangkat lunak harus menyediakan fasilitas untuk menyimpan data nomor, masa aktif dan status <i>simcard</i> yang	Melakukan penyimpanan data

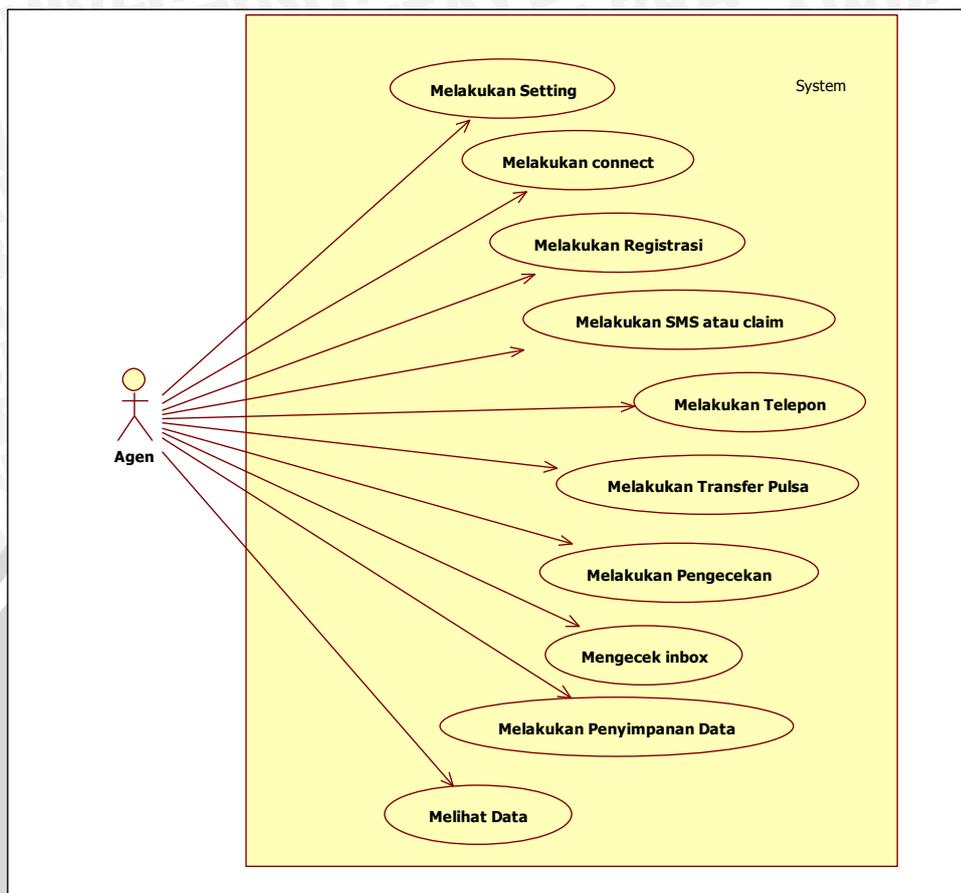
	telah diaktifkan ke basis data.	
SRS_010	Perangkat lunak harus mampu menampilkan data simcard yang telah di aktivasi dan mengetahui jumlah data yang telah di aktivasi sesuai dengan rentang tanggal yang telah ditentukan.	Melihat data

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 *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi yaitu diagram *use case* untuk agen.

Diagram *use case* pada perangkat lunak ini melibatkan agen sebagai aktor dan satu buah *use case*. Satu buah *use case* ini termasuk dalam aplikasi *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi. Satu buah *use case* ini juga akan disertai dengan skenario *use case* untuk menjelaskan rangkaian aktifitas yang terjadi di *use case* tersebut. Diagram *use case* aplikasi ini ditunjukkan dalam Gambar 4.5 berikut ini :





Gambar 4.4 Diagram Use Case Agen  
 Sumber : [Perancangan]

1. Skenario Use Case Melakukan Setting

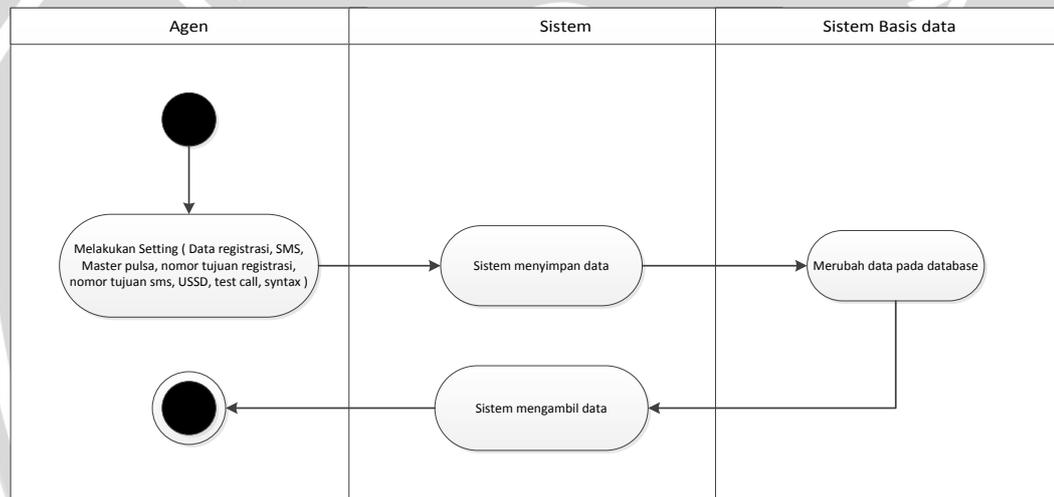
Tabel 4.3 Use case Melakukan Setting  
 Skenario Kasus Pada Sistem

Nomor Use Case	SRS_001
Nama	Melakukan Setting
Tujuan	Untuk melakukansettingdan menyimpan data setting
Deskripsi	Use Case ini menjelaskan bagaimana agen melakukan proses setting awal pada aplikasi dengan mengisi data registrasi, data sms, nomor tujuan, syntax transfer, dan USSD
Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak Activation Simcard sudah berjalan



	dan sudah terkoneksi dengan <i>modem pool</i> dan <i>database</i> aktifasi.
Aksi Aktor	Reaksi Sistem
Agan melakukan pengaturan atau <i>setting</i> untuk mengisi data yang akan di proses pada aplikasi	Sistem mengambil data data sebelumnya lalu melakukan penyimpanan data yang telah diubah ke dalam <i>database</i> aktifasi
Kondisi Akhir	Data yang telah dimasukkan di dalam <i>setting</i> akan tersimpan di <i>database</i> aktifasi

Sumber : [Perancangan]



Gambar 4.5 Diagram Aktifitas Melakukan *Setting*  
 Sumber : [Perancangan]

2. Skenario Use Case Melakukan *Connect*

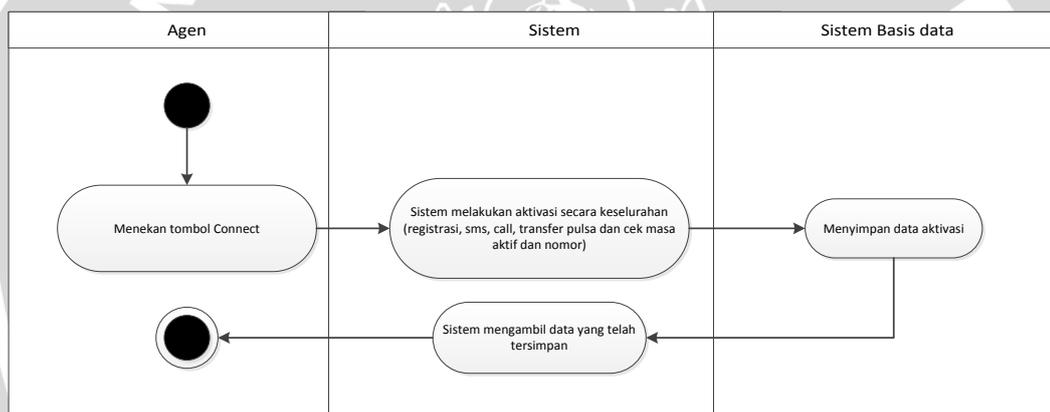
Tabel 4.4 Use Case Melakukan *Connect*

Skenario Kasus Pada Sistem	
Nomor <i>Use Case</i>	SRS_002
Nama	Melakukan <i>connect</i>
Tujuan	Untuk melakukan semua proses aktivasi untuk semua <i>simcard</i> tanpa harus satu persatu.
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana agen melakukan semua proses aktivasi <i>simcard</i> secara keseluruhan tanpa satu persatu.



Aktor	Agan
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Activation Simcard</i> sudah berjalan dan sudah terkoneksi dengan <i>modem pool</i> dan database aktivasi.
Aksi Aktor	Reaksi Sistem
Agan menekan tombol <i>connect</i> .	Sistem mengambil data pada <i>database</i> yang telah di <i>setting</i> lalu melakukan registrasi semua <i>simcard</i> dengan melakukan registrasi, sms, test call, <i>transfer pulsa</i> dan mengecek nomor dan masa aktif <i>simcard</i>
Kondisi Akhir	<i>Simcard</i> pada <i>modem pool</i> telah teraktivasi semua.

Sumber : [Perancangan]



Gambar 4.6 Diagram Aktifitas Melakukan *Connect*

Sumber : [Perancangan]

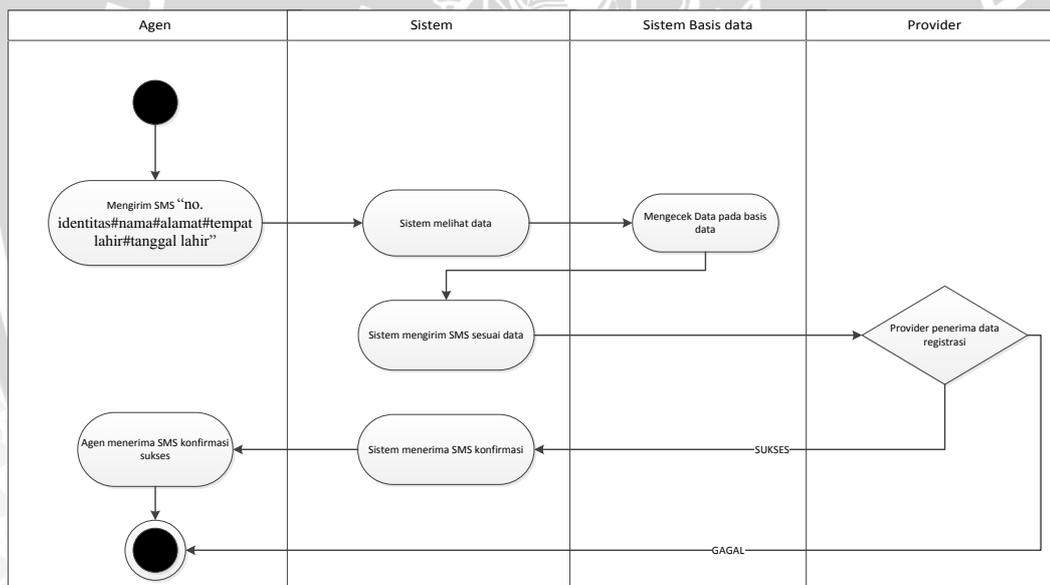
### 3. Skenario Use Case Melakukan Registrasi

Tabel 4.5 Use Case Melakukan Registrasi

Skenario Kasus Pada Sistem	
Nomor <i>Use Case</i>	SRS_003
Nama	Melakukan Registrasi Melalui SMS
Tujuan	Untuk melakukan registrasi melalui <i>SMS Gateway</i> .
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana agen melakukan proses registrasi <i>simcard</i> melalui <i>SMS Gateway</i> .

Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Activation Simcard</i> sudah berjalan dan sudah terkoneksi dengan <i>modem pool</i> dan <i>database</i> aktivasi.
Aksi Aktor	Reaksi Sistem
Agen SMS ke nomor <i>provider</i> dengan format “no. identitas#nama#alamat#tempat lahir#tanggal lahir”	Sistem mengambil data pada <i>database</i> yang telah di <i>setting</i>
Kondisi Akhir	Agen menerima SMS konfirmasi dari <i>provider</i> bahwa <i>simcard</i> sudah aktif atau sudah terdaftar

Sumber : [Perancangan]



Gambar 4.7 Diagram Aktifitas Melakukan Registrasi

Sumber : [Perancangan]

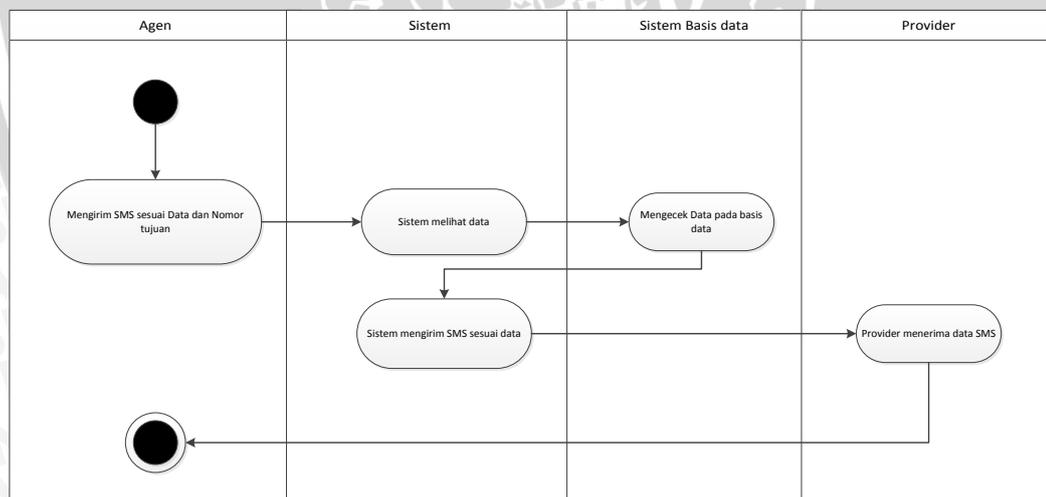
#### 4. Skenario Use Case Melakukan SMS

Tabel 4.6 Use Case Melakukan SMS

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_004
Nama	Melakukan SMS
Tujuan	Untuk melakukan SMS melalui <i>SMS Gateway</i> .

Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana agen melakukan proses SMS sesuai dengan data dan nomor tujuan melalui <i>SMS Gateway</i> .
Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Activation Simcard</i> sudah berjalan dan sudah terkoneksi dengan <i>modem pool</i> dan <i>database</i> aktivasi.
Aksi Aktor	Reaksi Sistem
Agan mengetikkan isi SMS yang akan dikirim ke nomor tujuan yang telah ditentukan	Sistem mengambil data SMS yang akan dikirim dan nomor sebagai tujuan SMS akan dikirimkan
Kondisi Akhir	Agan telah mengirim SMS ke nomor tujuan yang ditentukan

Sumber : [Perancangan]



Gambar 4.8 Diagram Aktifitas Melakukan SMS

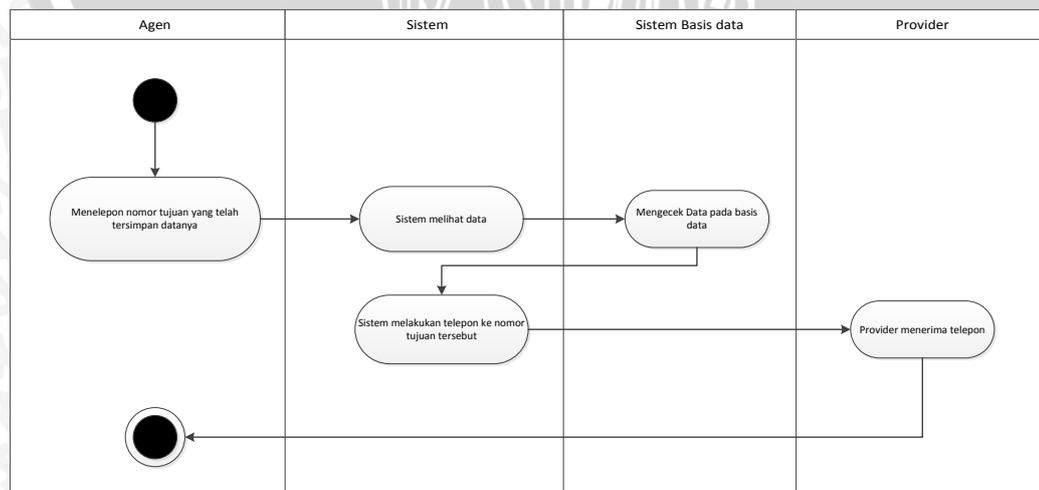
Sumber : [Perancangan]

5. Skenario Use Case Melakukan Telepon

Tabel 4.7 Use Case Melakukan Telepon

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_005
Nama	Melakukan telepon
Tujuan	Untuk melakukan telepon.
Deskripsi	Use Case ini menjelaskan bagaimana agen melakukan telepon ke nomor tujuan yang telah tersimpan datanya.
Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Activation Simcard</i> sudah berjalan dan sudah terkoneksi dengan <i>modem pool</i> dan <i>database</i> aktivasi.
Aksi Aktor	Reaksi Sistem
Agan mengetikkan nomor tujuan yang akan digunakan untuk nomor tujuan yang akan ditelepon.	Sistem mengambil data nomor tujuan yang telah dimasukkan dan melakukan telepon ke nomor tujuan tersebut.
Kondisi Akhir	Agan melakukan telepon ke nomor tujuan yang diinginkan.

Sumber : [Perancangan]



Gambar 4.9 Diagram Aktifitas Melakukan Telepon

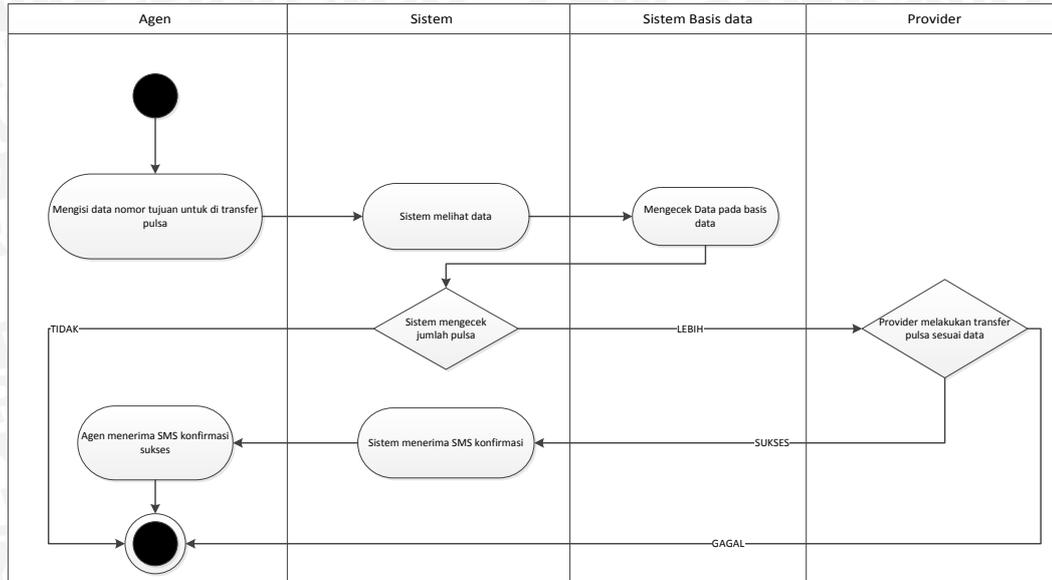
Sumber : [Perancangan]

## 6. Skenario Use Case Melakukan Transfer Pulsa

Tabel 4.8 Use Case Melakukan Transfer Pulsa

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_006
Nama	Melakukan <i>transfer</i> pulsa
Tujuan	Untuk melakukan <i>transfer</i> pulsa
Deskripsi	<i>Use Case</i> ini menjelaskan bagaimana agen melakukan <i>transfer</i> ke nomor yang telah ditentukan apabila <i>simcard</i> memiliki pulsa lebih
Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Activation Simcard</i> sudah berjalan dan sudah terkoneksi dengan <i>modem pool</i> dan database aktivasi.
Aksi Aktor	Reaksi Sistem
Agen mengetikkan nomor tujuan yang akan digunakan untuk nomor tujuan <i>transfer</i> pulsa apabila ada pulsa lebih	Sistem mengecek pulsa dan apabila ada kelebihan pulsa maka pulsa di <i>transfer</i> ke nomor yang telah di ketik.
Kondisi Akhir	Agen telah melakukan <i>transfer</i> pulsa ke nomor yang ditentukan apabila ada kelebihan pulsa

Sumber : [Perancangan]



Gambar 4.10 Diagram Aktifitas Melakukan Transfer Pulsa  
 Sumber : [Perancangan]

7. Skenario Use Case Melakukan Pengecekan

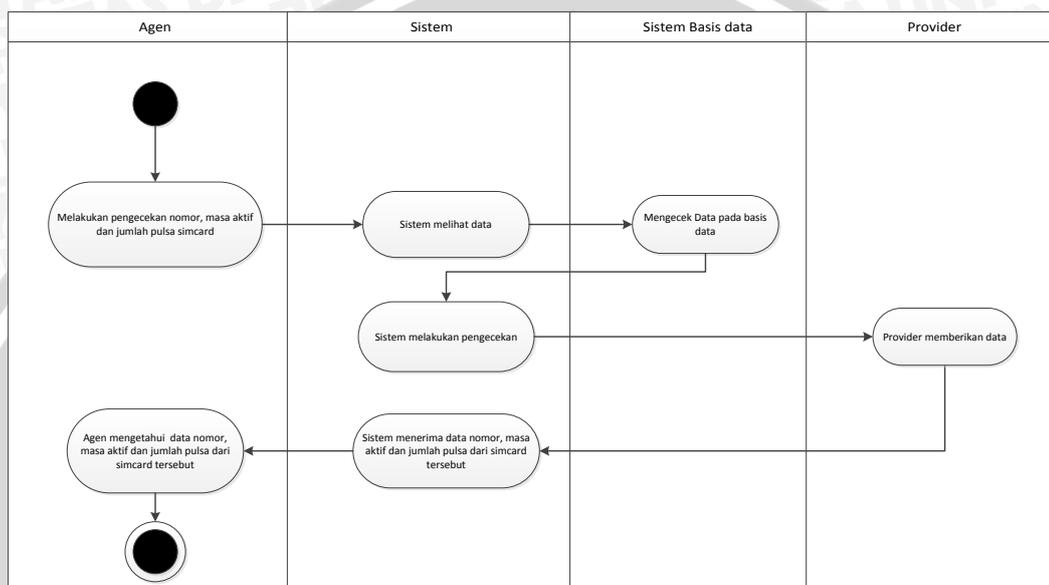
Tabel 4.9 Use Case Melakukan Pengecekan

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_007
Nama	Melakukan pengecekan
Tujuan	Untuk melakukan pengecekan.
Deskripsi	Use Case ini menjelaskan bagaimana agen melakukan pengecekan nomor, masa aktif dan jumlah pulsa dari simcard.
Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak Activation Simcard sudah berjalan dan sudah terkoneksi dengan modem pool dan database aktivasi.
Aksi Aktor	Reaksi Sistem
Agen memasukkan nomor untuk pengecekan menelepon untuk mengetahui nomor,	Sistem mengambil data nomor untuk pengecekan lalu melakukan telepon ke nomor tersebut lalu menerima data dari provider.



masa aktif dan jumlah pulsa <i>simcard</i> .	
Kondisi Akhir	Agen dapat mengetahui nomor <i>simcard</i> tersebut dan mengetahui masa aktif <i>simcard</i> tersebut apabila telah aktif dan jumlah pulsa <i>simcard</i> tersebut.

Sumber : [Perancangan]



Gambar 4.11 Diagram Aktifitas Melakukan Pengecekan

Sumber : [Perancangan]

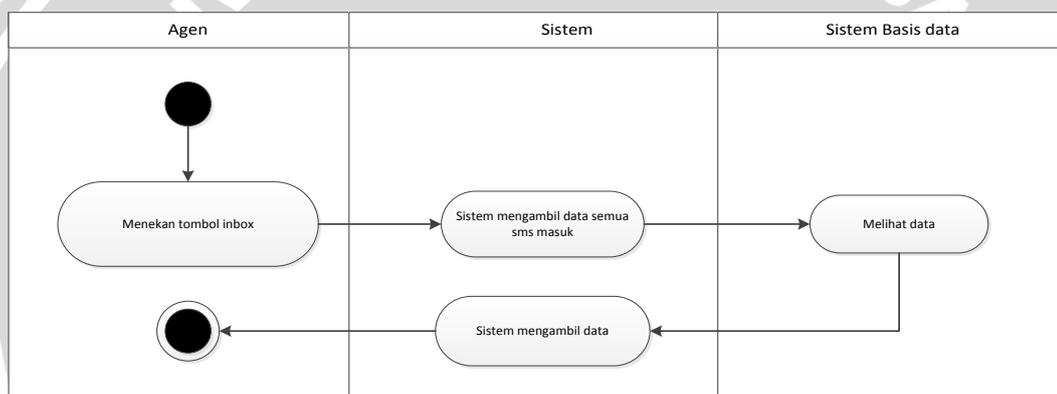
## 8. Skenario Use Case Mengecek Inbox

Tabel 4.10 Use Case Mengecek Inbox

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_008
Nama	Mengecek <i>inbox</i>
Tujuan	Untuk mengecek <i>inbox</i> .
Deskripsi	Use Case ini untuk mengecek <i>inbox</i> bila ada sms masuk ke <i>simcard</i> telah mendapatkan laporan registrasi berhasil.
Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Activation Simcard</i> sudah berjalan dan sudah terkoneksi dengan <i>modem pool</i> dan

	<i>database</i> aktivasi.
Aksi Aktor	Reaksi Sistem
Agen melakukan pengecekan sms balasan registrasi dari <i>provider</i> pada <i>inbox simcard</i> .	Sistem mengambil data sms pada <i>database</i> .
Kondisi Akhir	Agen mengetahui status <i>simcard</i> telah aktif apabila telah mendapat sms telah melakukan registrasi dari <i>provider</i> .

Sumber : [Perancangan]



Gambar 4.12 Diagram Aktifitas Mengecek *Inbox*

Sumber : [Perancangan]

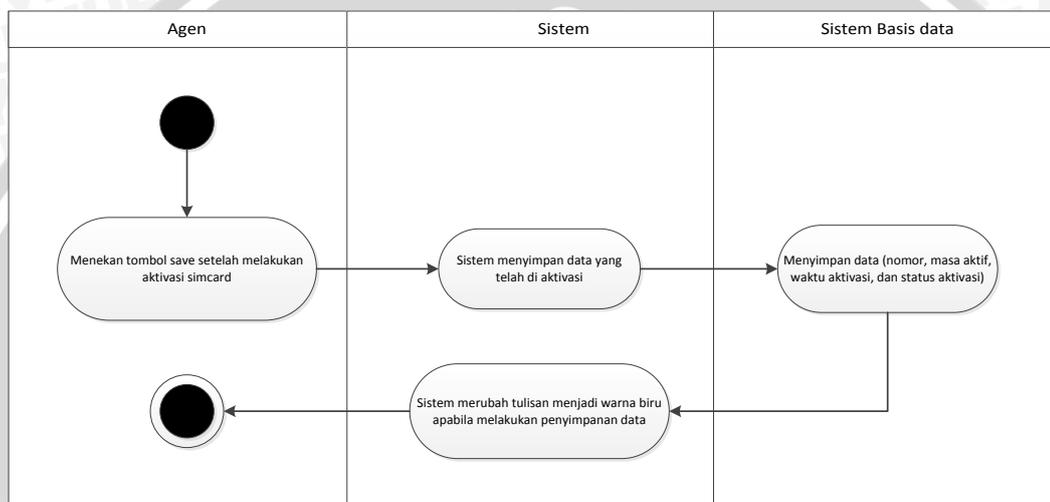
## 9. Skenario Use Case Melakukan Penyimpanan Data

Tabel 4.11 Use Case Melakukan Penyimpanan Data

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_009
Nama	Melakukan penyimpanan data
Tujuan	Untuk melakukan penyimpanan data.
Deskripsi	Use Case ini menjelaskan bagaimana agen melakukan penyimpanan data ke basis data.
Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Activation Simcard</i> sudah berjalan

	dan sudah terkoneksi dengan <i>database</i> aktivasi dan telah melakukan aktivasi <i>simcard</i> .
Aksi Aktor	Reaksi Sistem
Agen menekan tombol <i>save</i> .	Sistem menyimpan data aktivasi berupa nomor, masa aktif dan waktu aktivasi.
Kondisi Akhir	Agen dapat menyimpan data <i>simcard</i> yang telah di aktivasi.

Sumber : [Perancangan]



Gambar 4.13 Diagram Aktifitas Melakukan Penyimpanan Data

Sumber : [Perancangan]

10. Skenario Use Case Melihat Data

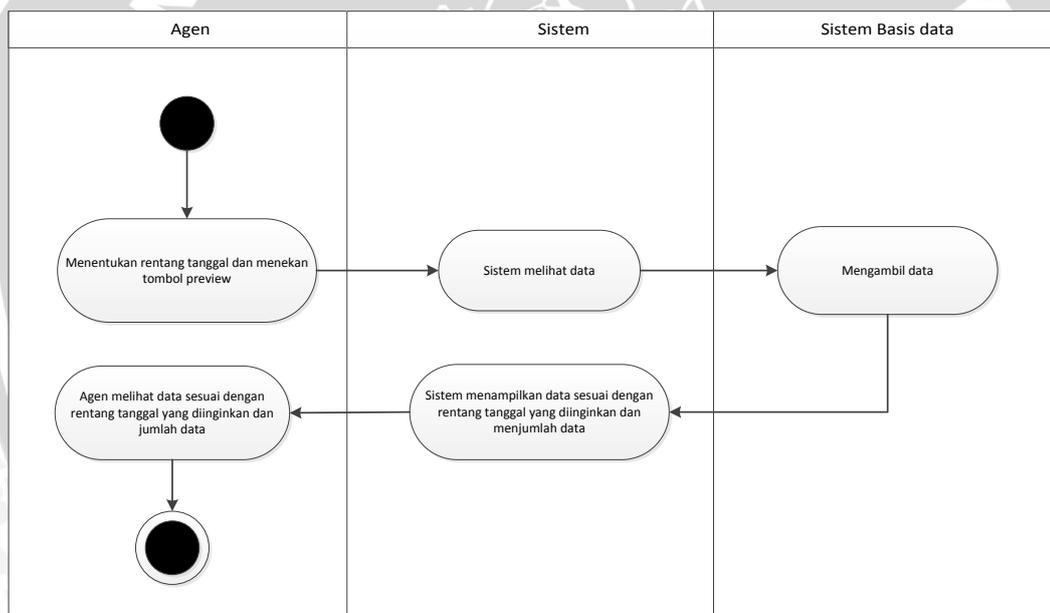
Tabel 4.12 Use Case Melihat Data Skenario Kasus Pada Sistem

Nomor Use Case	SRS_010
Nama	Melihat Data
Tujuan	Untuk melihat data yang telah tersimpan.
Deskripsi	Use Case ini menjelaskan bagaimana agen dapat melihat semua data <i>simcard</i> yang telah di aktivasi dan dapat mengetahui jumlah <i>simcard</i> yang telah di aktivasi sesuai rentang tanggal yang ditentukan.
Aktor	Agen
Skenario Utama	
Kondisi Awal	Perangkat lunak <i>Activation Simcard</i> sudah berjalan



	dan sudah terkoneksi dengan <i>database</i> aktivasi dan telah melakukan aktivasi simcard.
Aksi Aktor	Reaksi Sistem
Agen menekan data lalu menentukan rentang tanggal dan setelah itu menekan tombol <i>preview</i> .	Sistem menampilkan data berupa nomor, masa aktif dan status sesuai dengan rentang tanggal yang telah di tentukan. Sistem juga akan menjumlah data yang di tampilkan.
Kondisi Akhir	Agen dapat melihat data dan mengetahui jumlah <i>simcard</i> yang telah di aktivasi sesuai dengan rentang tanggal yang telah ditentukan.

Sumber : [Perancangan]



Gambar 4.14 Diagram Aktifitas Melihat Data  
 Sumber : [Perancangan]

#### 4.1.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak memiliki tiga tahap, yaitu perancangan basis data, perancangan algoritma, dan perancangan antarmuka pengguna dari perangkat lunak *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi.



#### 4.1.2.1 Perancangan Basis Data

Basis data berfungsi sebagai tempat menyimpan data. Perancangan basis data digunakan untuk merancang basis data yang akan dibuat agar masukan dan keluaran program sesuai dengan apa yang diharapkan. Perancangan basis data mengambil acuan dari proses analisis data yang dilakukan pada tahap analisis kebutuhan.



**Gambar 4.15 Diagram Entity Relationship Perangkat Lunak Activation Simcard dengan Menggunakan Modem Pool Sebagai Alat Aktivasi**  
Sumber : [Perancangan]

Berikut ini merupakan struktur tabel serta keterangan masing masing tabel dan *field* yang ada pada *database*. Tabel aplikasi *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi ini memiliki 2 tabel yaitu tabel *dataact* dan *sett\_app*. Struktur tabel *dataact* ditunjukkan dalam Tabel 4.13.

**Tabel 4.13 Struktur tabel *dataact***

No.	Nama Field	Tipe	Lebar	Keterangan
1	id	Integer	10	<i>Id</i> aktivasi.
2	MSISDN	Varchar	20	Nomor <i>simcard</i> .
4	MasaAktif	Vachar	50	Masa aktif <i>simcard</i> .
5	WaktuAct	Date time		Waktu melakukan aktivasi.
6	StatusAct	Varchar	160	Status <i>simcard</i> .

Sumber : [Perancangan]

Entitas *sett\_app* merepresentasikan tabel setting, yang merupakan tabel yang berisi *setting* dari aplikasi yaitu nama *setting*, isi *setting* dan tujuan. Struktur tabel *sett\_app* ditunjukkan dalam Tabel 4.14.

**Tabel 4.14 Struktur tabel *sett\_app***

No.	Nama Field	Tipe	Lebar	Keterangan
1	id	Integer	11	<i>Id</i> .

2	nama_setting	Varchar	255	Nama data yang akan di <i>setting</i> .
3	Isi_setting	Varchar	255	Isi data yang akan di <i>setting</i> .
4	tujuan	Varchar	20	Nomor tujuan yang akan digunakan dalam melakukan registrasi dan sms.

Sumber : [Perancangan]

#### 4.1.2.2 Perancangan Algoritma

Perancangan algoritma aplikasi *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi ini memiliki dua proses perancangan algoritma yaitu perancangan algoritma halaman utama dan perancangan algoritma *setting*. Pada perancangan algoritma halaman utama memiliki 9 proses dimana salah satunya merupakan perancangan algoritma *setting*.

##### a. Perancangan Algoritma Halaman *Setting*

Proses pada halaman *setting* bertujuan untuk melakukan pengaturan sebelum melakukan proses yang ada pada halaman utama. Dimana akan ada untuk pengisian data registrasi, sms, nomor untuk *transfer* pulsa, nomor untuk test *call*, syntax untuk *transfer* pulsa, dan nomor untuk mengecek masa aktif. Perancangan algoritma *setting* aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_001.

**NAMA ALGORITMA** Halaman Setting

**DESKRIPSI**

Masukan : Menekan tombol setting

Proses :

1. Mengisi data registrasi pada kolom "Data registrasi".
2. Mengisi data sms pada kolom "SMS".
3. Mengisi data nomor tujuan transfer pulsa pada kolom "Master Transfer Pulsa".
4. Mengisi data nomor untuk tujuan telepon pada kolom "Test Call".
5. Mengisi data syntax untuk syntax transfer pulsa pada kolom "Transfer Syntax".
6. Mengisi data nomor untuk tujuan registrasi pada kolom "Tujuan".
7. Mengisi data nomor untuk tujuan SMS pada kolom "Tujuan".
8. Mengisi data nomor untuk mengecek nomor dan masa aktif simcard pada kolom "USSD".

9. Menekan Tombol "Save".
  10. Sistem akan menyimpan data yang telah dimasukkan ke dalam database.
- Keluaran :
1. Halaman utama muncul di layar dan data telah tersimpan.

**Gambar 4.16 Perancangan Algoritma Halaman Setting**  
**Sumber: [Perancangan]**

### b. Perancangan Algoritma Connect

Proses pada algoritma *connect* bertujuan untuk melakukan proses aktivasi secara keseluruhan mulai dari registrasi, *call*, *transfer* pulsa, sms dan mengecek masa aktif *simcard*. Perancangan algoritma *connect* aplikasi *Activation Simcard* mengacu pada spesifikasi kebutuhan SRS\_002.

**NAMA ALGORITMA** Connect

**DESKRIPSI**

Masukan : Menekan tombol connect

Proses :

1. Sistem menjalankan "TimerUtama".
2. Timer utama akan memberikan selisih waktu 2 detik untuk setiap port.
3. Sistem mengatur modem ke mode text dengan 'AT+CMGF=1'.
4. Sistem mengatur media penyimpanan ke *simcard* dengan 'AT+CPMS="SM"'.
5. Sistem mengatur karakter ke GSM dengan 'AT+CSCS="GSM"'.
6. Sistem melakukan registrasi ke jaringan dengan 'AT+CREG=1'.
7. Sistem melakukan *unlock simcard* dengan 'AT+CLCK="FD,0,1234"'.
8. Sistem melakukan telepon dengan 'ATD'.
9. Sistem melakukan penutupan telepon dengan 'ATH'.
10. Sistem melakukan proses sms dengan 'AT+CMGS'.
11. Sistem melakukan proses registrasi dengan 'AT+CMGS'.
12. Sistem melakukan pengecekan dengan 'AT+CUSD=1'.

Keluaran :

1. *Simcard* telah aktif.

**Gambar 4.17 Perancangan Algoritma Connect**  
**Sumber: [Perancangan]**

### c. Perancangan Algoritma Registrasi

Proses pada algoritma registrasi bertujuan untuk melakukan proses registrasi dengan mengirim data registrasi yang berisi nomor identitas, nama,

alamat, tempat dan tanggal lahir ke 4444. Perancangan algoritma registrasi aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_003.

**NAMA ALGORITMA** Registrasi

**DESKRIPSI**

Masukan : Menekan tombol registrasi

Proses :

1. Sistem mengambil data registrasi pada database.
2. Sistem melakukan proses registrasi dengan melakukan sms gateway dengan perintah 'AT+CMGS'.

Keluaran :

1. *Simcard* telah melakukan registrasi.

**Gambar 4.18 Perancangan Algoritma Registrasi**

Sumber: [Perancangan]

#### d. Perancangan Algoritma SMS

Proses pada algoritma sms bertujuan untuk melakukan proses sms dengan mengirim data sms ke nomor tujuan yang telah ditentukan. Perancangan algoritma sms aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_004.

**NAMA ALGORITMA** SMS

**DESKRIPSI**

Masukan : Menekan tombol sms

Proses :

1. Sistem mengambil data sms pada database.
2. Sistem melakukan proses sms dengan melakukan msgatewaydengan perintah 'AT+CMGS'.

Keluaran :

1. *Simcard* telah melakukan sms.

**Gambar 4.19 Perancangan Algoritma SMS**

Sumber: [Perancangan]

#### e. Perancangan Algoritma Telepon

Proses pada algoritma telepon bertujuan untuk melakukan telepon ke nomor tujuan yang telah ditentukan yang telah disimpan pada *database*.

Perancangan algoritma telepon aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_005.

**NAMA ALGORITMA** Telepon

**DESKRIPSI**

Masukan : Menekan tombol call

Proses :

1. Sistem mengambil data nomor tujuan untuk telepon pada database.
2. Sistem melakukan proses telepon dengan perintah 'ATD[nomor tujuan];'.
3. Dalam waktu yang telah ditentukan sistem akan menutup telepon dengan perintah 'ATH'.

Keluaran :

1. *Simcard* telah melakukan telepon.

**Gambar 4.20 Perancangan Algoritma Telepon**

Sumber: [Perancangan]

#### f. Perancangan Algoritma *Transfer Pulsa*

Proses pada algoritma *transfer pulsa* bertujuan untuk melakukan proses *transfer pulsa* ke nomor tujuan yang telah ditentukan apabila ada kelebihan pulsa. Perancangan algoritma *transfer pulsa* aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_006.

**NAMA ALGORITMA** SMS

**DESKRIPSI**

Masukan : Melakukan pengecekan jumlah pulsa

Proses :

1. Sistem melakukan pengecekan jumlah pulsa.
2. Sistem melakukan transfer pulsa apabila ada kelebihan pulsa dan di transfer ke nomor tujuan yang telah ditentukan.

Keluaran :

1. Pulsa telah di transfer apabila ada kelebihan pulsa.

**Gambar 4.21 Perancangan Algoritma *Transfer Pulsa***

Sumber: [Perancangan]

### g. Perancangan Algoritma Pengecekan

Proses pada algoritma pengecekan bertujuan untuk melakukan proses pengecekan nomor *simcard* dan masa aktif *simcard*. Perancangan algoritma cek aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_007.

**NAMA ALGORITMA** Pengecekan

**DESKRIPSI**

Masukan : Menekan tombol cek

Proses :

1. Sistem melakukan pengecekan dengan perintah 'AT+CUSD=1'.
2. Sistem mengambil data nomor dari *simcard* dan masa aktif *simcard*.

Keluaran :

1. Mendapatkan data nomor *simcard* dan masa aktif *simcard*.

**Gambar 4.22 Perancangan Algoritma Pengecekan**  
Sumber: [Perancangan]

### h. Perancangan Algoritma Mengecek *Inbox*

Proses pada algoritma mengecek *inbox* bertujuan untuk membaca sms masuk setelah melakukan registrasi. Perancangan algoritma mengecek *inbox* aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_008.

**NAMA ALGORITMA Mengecek** Inbox

**DESKRIPSI**

Masukan : Menekan tombol inbox

Proses :

1. Sistem membaca semua sms yang telah masuk dengan perintah 'AT+CMGL=ALL'.
2. Sistem mengambil data sms masuk.

Keluaran :

1. Mendapatkan data semua sms masuk.

**Gambar 4.23 Perancangan Algoritma Mengecek *Inbox***  
Sumber: [Perancangan]

### i. Perancangan Algoritma *Save*

Proses pada algoritma *save* bertujuan untuk melakukan proses penyimpanan data nomor, masa aktif, waktu aktivasi dan status aktivasi ke

*database*. Perancangan algoritma *save* aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_009.

**NAMA ALGORITMA** Save

**DESKRIPSI**

Masukan : Menekan tombol save

Proses :

1. Sistem mengambil data nomor dan menyimpannya pada database dan mengubah warna tulisan menjadi biru.
2. Sistem mengambil data masa aktif dan menyimpannya pada database dan mengubah warna tulisan menjadi biru.
3. Sistem mengambil data status dan menyimpannya pada database dan mengubah warna tulisan menjadi biru.

Keluaran :

1. Data telah tersimpan pada database.

**Gambar 4.24 Perancangan Algoritma Save**  
Sumber: [Perancangan]

#### j. Perancangan Algoritma Data

Proses pada algoritma data bertujuan untuk melihat data aktivasi *simcard* yang telah tersimpan di *database*. Perancangan algoritma data aplikasi *activation simcard* mengacu pada spesifikasi kebutuhan SRS\_010.

**NAMA ALGORITMA** Data

**DESKRIPSI**

Masukan : Menekan tombol data

Proses :

1. Sistem menampilkan form data.
2. Sistem mengambil data pada database sesuai tanggal yang telah ditentukan.
3. Sistem menjumlah data yang ditampilkan.

Keluaran :

1. Data sesuai tanggal yang ditentukan dan jumlah data yang tampil.

**Gambar 4.25 Perancangan Algoritma Data**  
Sumber: [Perancangan]

#### 4.1.2.3 Perancangan Antarmuka

Pada bagian ini akan dijelaskan tentang perancangan antarmuka perangkat lunak *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi.

Antarmuka perangkat lunak ini akan digunakan oleh pengguna untuk berinteraksi dengan sistem perangkat lunak *Activation Simcard*. Antarmuka perangkat lunak ini dibagi menjadi dua, yaitu antarmuka halaman utama dan antarmuka halaman *setting*.

#### a. Perancangan Antarmuka Halaman Utama

Antarmuka pengguna untuk *Activation Simcard* berupa sebuah halaman *desktop*. Antarmuka halaman utama mencakup semua kebutuhan untuk aktivasi yang menampilkan semua *groupbox* atau *port* dan data yang akan disimpan.

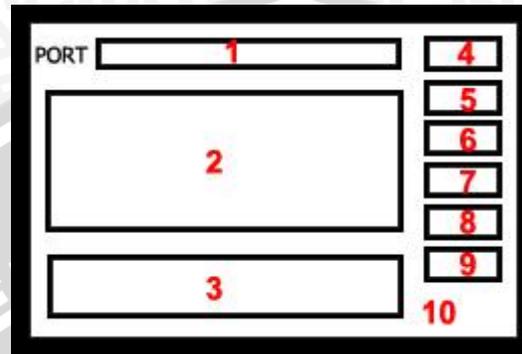
The screenshot shows a web-based interface for simcard activation. At the top, there are several buttons labeled '1-4', '5-8', '9-12', '13-16', '17-20', and '21-24'. A red '1' is placed over the '9-12' button. To the right, there is a 'PORT SETTING [ 1 - 24 ]' section with a red '2' over a dropdown menu. Below these are four empty rectangular boxes, with a red '3' in the top-left one. To the right of these boxes is a larger empty rectangular area with a red '4' in the center. At the bottom left, there are four rows of input fields, each with a red number: '5' over the 'Nojop' field, '6' over the 'Aktif' checkbox, and '7' over the 'Status' field. At the bottom right, there are four buttons labeled '8', '9', '10', and '11', with a red '12' below them.

Gambar 4.26 Tampilan Antarmuka Halaman Utama  
Sumber: [Perancangan]

Gambar 4.27 memiliki keterangan sebagai berikut :

1. Tombol *Port* untuk memilih port yang akan ditampilkan
2. *Field Port Setting* untuk menentukan jumlah *port* yang akan digunakan.
3. *Field Line* untuk menampilkan proses pada satu *port*.
4. *MemoError* untuk menampilkan peringatan bila terjadi *error* pada suatu *port*.
5. *Field Nomor* untuk menampilkan nomor.
6. *Field Aktif* untuk menampilkan masa aktif.
7. *Field Status* untuk menampilkan sms balasan registrasi.
8. Tombol *Connect* untuk menjalankan proses *connect* keseluruhan.

9. Tombol *Save* untuk menjalankan proses *save*.
10. Tombol *Data* untuk menjalankan proses *save*.
11. Tombol *Setting* untuk menjalankan proses *setting*.
12. *FieldTimer* untuk menampilkan *timer*.



Gambar 4.27 Tampilan Antarmuka *Groupbox*  
Sumber: [Perancangan]

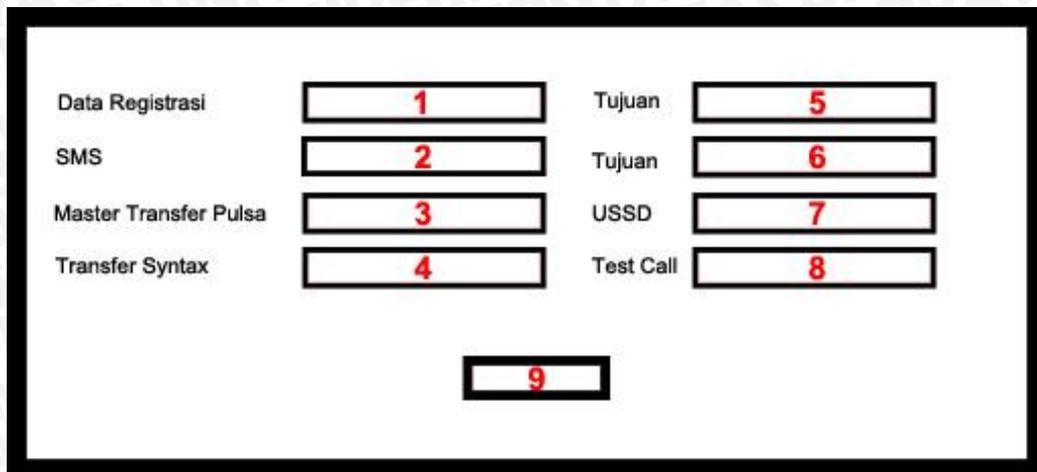
Gambar 4.28 memiliki keterangan sebagai berikut :

1. Tombol *Port* untuk memilih *port*.
2. *Field Memo* untuk menampilkan keterangan proses dan hasil proses.
3. *Field Memo* untuk menampilkan *AT Command* dan respon *modem*.
4. Tombol *Conn* untuk menjalankan proses *connect* pada *groupbox* tersebut.
5. Tombol *Reg* untuk menjalankan proses registrasi.
6. Tombol *SMS* untuk menjalankan proses SMS.
7. Tombol *Cek* untuk menjalankan proses pengecekan nomor dan masa aktif.
8. Tombol *Call* untuk menjalankan proses telepon.
9. Tombol *Inbox* untuk menjalankan proses pengecekan *inbox*.
10. *FieldTimer* untuk menampilkan *timer* pada *groupbox* tersebut.

#### b. Perancangan Antarmuka Halaman *Setting*

Antarmuka halaman *setting* digunakan untuk menyimpan atau mengedit data-data yang akan digunakan pada aplikasi seperti data registrasi, nomor

tujuan registrasi, data sms, nomor tujuan sms, nomor tujuan *transfer* pulsa, nomor tujuan telepon, *USSD*, *syntax* dan *port setting*.



The image shows a settings interface with two columns of input fields. The left column contains four fields labeled 'Data Registrasi', 'SMS', 'Master Transfer Pulsa', and 'Transfer Syntax', numbered 1 through 4. The right column contains three fields labeled 'Tujuan', 'Tujuan', 'USSD', and 'Test Call', numbered 5 through 8. Below these fields is a single button labeled '9'.

Data Registrasi	1	Tujuan	5
SMS	2	Tujuan	6
Master Transfer Pulsa	3	USSD	7
Transfer Syntax	4	Test Call	8
		9	

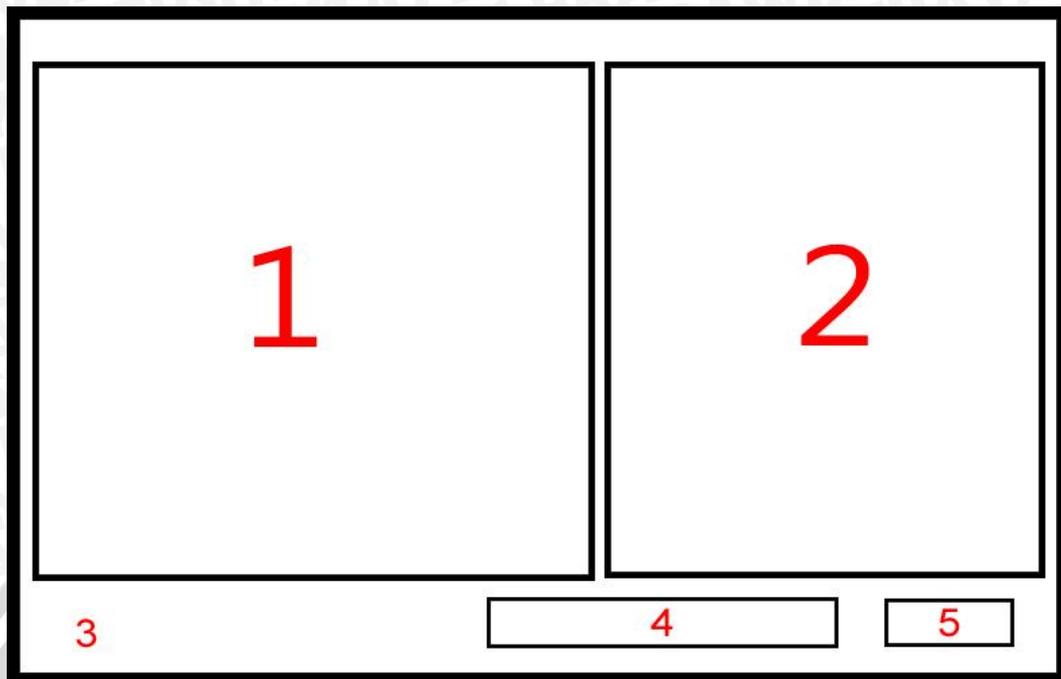
Gambar 4.28 Tampilan Antarmuka Halaman *Setting*  
Sumber: [Perancangan]

Gambar 4.29 memiliki keterangan sebagai berikut :

1. *Field* Data Registrasi untuk mengisi data registrasi.
2. *Field* SMS untuk mengisi data SMS.
3. *Field* *Master Transfer Pulsa* untuk mengisi nomor tujuan *transfer* pulsa.
4. *Field* *Transfer Syntax* untuk mengisi *syntax* yang digunakan untuk *transfer* pulsa.
5. *Field* Tujuan untuk mengisi nomor tujuan registrasi.
6. *Field* Tujuan untuk mengisi nomor tujuan SMS.
7. *Field* *USSD* untuk mengisi *syntax* untuk mengecek nomor dan masa aktif.
8. *Field* *Test Call* untuk mengisi nomor tujuan telepon.
9. Tombol *Save* untuk menjalankan proses *save*.

### c. Perancangan Antarmuka Halaman Data

Antarmuka halaman data digunakan untuk menampilkan data yang telah tersimpan pada *database* dan menampilkan jumlah data yang telah ditentukan.



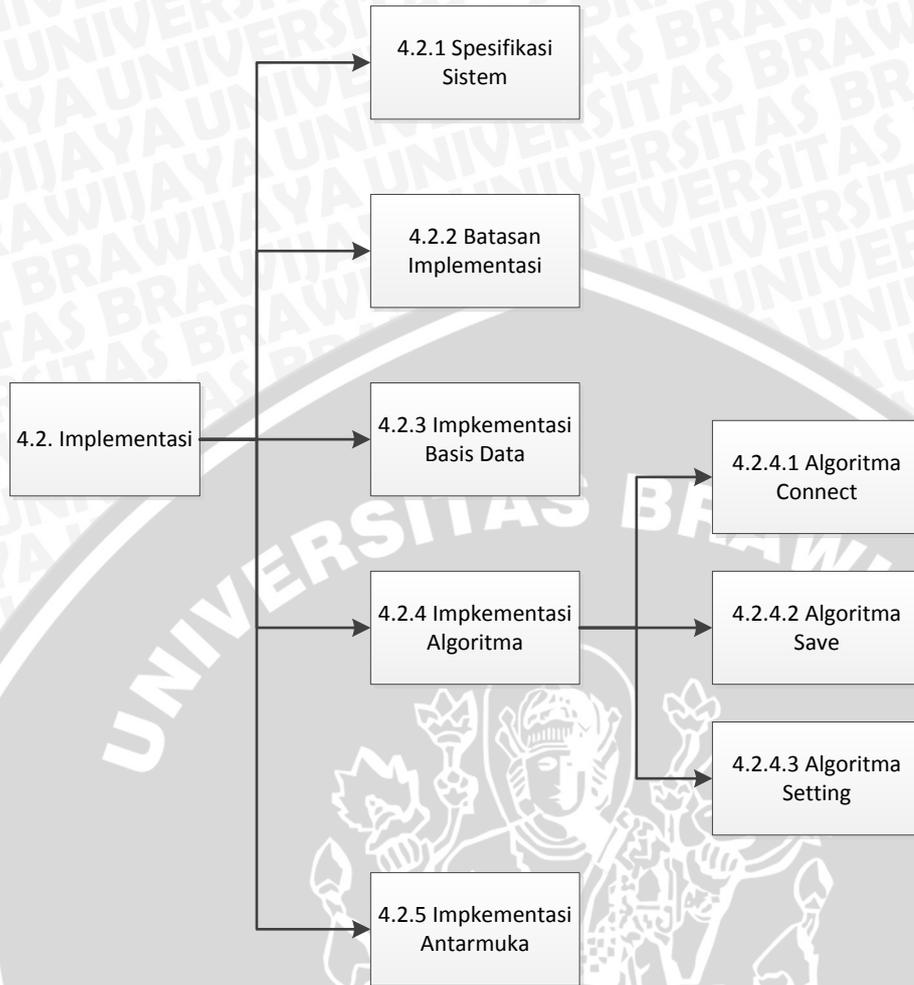
Gambar 4.29 Tampilan Antarmuka Halaman Data  
Sumber: [Perancangan]

Gambar 4.30 memiliki keterangan sebagai berikut :

1. *Field* Data untuk menampilkan data nomor, masa aktif, waktu aktivasi, dan status aktivasi.
2. *Field* Grafik untuk menampilkan grafik.
3. *Field* Jumlah untuk menampilkan jumlah data yang ditampilkan.
4. *Field* Tanggal untuk menentukan tanggal data yang akan ditampilkan.
5. Tombol *Preview* untuk menampilkan data.

#### 4.2 Implementasi

Sub bab ini membahas mengenai tahapan implementasi perangkat lunak *Activation Simcard* menggunakan *modem pool* berdasarkan hasil yang telah didapatkan dari analisis kebutuhan dan proses perancangan perangkat lunak. Pembahasan terdiri atas penjelasan tentang spesifikasi sistem, batasan – batasan dalam implementasi, implementasi basis data, implementasi algoritma, dan implementasi antarmuka.



Gambar 4.30 Diagram Blok Implementasi  
 Sumber : [Implementasi]

**4.2.1 Spesifikasi Sistem**

Aplikasi *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi dikembangkan dalam lingkungan implementasi yang terdiri dari perangkat keras dan perangkat lunak.

**4.2.1.1 Spesifikasi Perangkat Keras**

Spesifikasi perangkat keras yang dipakai dalam proses pengembangan dijelaskan dalam Tabel 4.15.

Tabel 4.15 Spesifikasi Perangkat Keras Komputer  
**Notebook Acer Aspire 4736**

<i>Processor</i>	Intel (R) Core (TM)2 Duo CPU T6600 2.2 GHz
<i>Memory (RAM)</i>	3.00 GB



<i>Harddisk</i>	320 GB Serial ATA 5400 RPM
<i>Motherboard</i>	Intel Pentium Motherboard
<i>Monitor</i>	14.1" WXGA LED
<b>Modem pool Simbox Wavecom</b>	
<i>Model</i>	Q2406B
<i>Jenis</i>	GSM/GPRS:900/1800/1900MHZ

Sumber : [Implementasi]

#### 4.2.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang dipakai dalam proses pengembangan aplikasi *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi dijelaskan dalam Tabel 4.16.

**Tabel 4.16 Spesifikasi Perangkat Lunak Komputer**

<b>Notebook Acer Aspire 4736</b>	
<i>Operating System</i>	Microsoft Windows 7 32-bit
<i>Programming Language</i>	Delphi
<i>Database Management System</i>	MySQL 4.1
<i>Integrated Development Environment</i>	Embarcadero RAD Studio XE2
<i>Software Development Tools</i>	SQLyog Enterprise 8.18

Sumber : [Implementasi]

#### 4.2.2 Batasan Implementasi

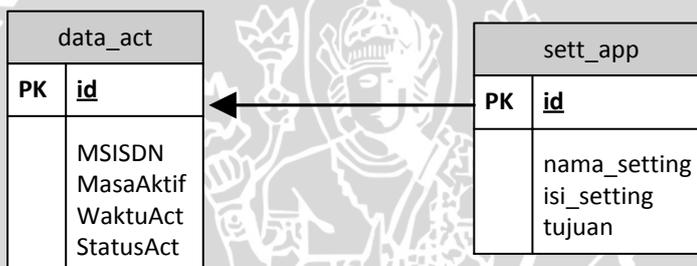
Beberapa batasan dalam mengimplementasikan perangkat lunak *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi sebagai berikut :

1. Perangkat lunak *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi dirancang dan dijalankan dengan menggunakan *Delphi Desktop Application*.
2. *Database Management System* yang digunakan adalah MySQL.

3. Perangkat lunak *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi menggunakan tarif SMS, tarif telepon, dan nomor *handphone* regular.
4. Perangkat lunak *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi menggunakan 1 basis data server.

### 4.2.3 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 4.32 menggambarkan diagram konseptual *entitiy relationship* dari perangkat lunak *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi.



Gambar 4.31 Diagram ER Konseptual dari Perangkat Lunak *Activation Simcard* Menggunakan *Modem Pool* Sebagai Alat Aktivasi  
 Sumber : [Implementasi]

### 4.2.4 Implementasi Algoritma

Aplikasi *Activation Simcard* menggunakan *modem pool* mempunyai beberapa proses (*method*) utama. Pada penulisan skripsi ini hanya dicantumkan algoritma dari beberapa proses saja sehingga tidak semua implementasi algoritma *method* akan dicantumkan. Implementasi algoritma proses yang dicantumkan antara lain adalah proses *connect*, proses *save*, proses *setting*, dan proses data. Implementasi algoritma ini akan direpresentasikan dalam bentuk *code* dengan bahasa pemrograman Delphi.

#### 4.2.4.1 Implementasi Algoritma Connect

Operasi pada proses *connect* yang ada di halaman utama bertujuan untuk melakukan aktivasi *simcard*. Proses *connect* melakukan semua proses aktivasi mulai dari *registrasi*, *call* dan *transfer* pulsa secara keseluruhan dengan

menggunakan *timer* dalam melakukan proses satu persatu. Gambar 4.33 merupakan *proceduredari timer* setiap *groupbox* yang berjalan ketika proses *connect* dilakukan.

```
1. procedure TForm1.Timer1Timer(Sender: TObject);
2. begin
3.   N1.Font.Color := clBlack;
4.   A1.Font.Color := clBlack;
5.   S1.Font.Color := clBlack;
6.   LT1.Caption := IntToStr(StrToInt(LT1.Caption)+1);
7.   if LT1.Caption = '1' then
8.     begin
9.       Mem1.Text := Mem1.Text + 'Connecting to Modem';
10.      Mem1.Text := Mem1.Text + #13#10 + 'Sett modem to
11.        mode text';
12.      ComPort1.WriteStr('AT+CMGF=1'+ #13#10);
13.      Mem1.Text := Mem1.Text + #13#10 + 'Sett storage
14.        to SimCard';
15.      ComPort1.WriteStr('AT+CPMS="SM"'+#13#10);
16.      Mem1.Text := Mem1.Text + #13#10 + 'Sett character
17.        to GSM';
18.      ComPort1.WriteStr('AT+CSCS="GSM"'+ #13#10);
19.      memo2.Clear;
20.      mem1.Text := mem1.Text + #13#10 + 'Register to
21.        Network';
22.      Comport1.WriteStr('AT+CREG=1'+#13#10);
23.      memo2.Clear;
24.      mem1.Text := mem1.Text + #13#10 + 'Unlock
25.        Simcard';
26.      Comport1.WriteStr('AT+CLCK="FD",0,1234'+#13#10);
27.      mem1.Text := mem1.Text + #13#10 + 'Ready for
28.        activation process';
29.    end;
30.    if LT1.Caption = '2' then
31.      begin
32.        memo2.Clear;
33.        mem1.Text := mem1.Text + #13#10 + 'Call ' +
34.          string(form2.ucall.Text);
35.        Comport1.WriteStr('ATD' + string(form2.ucall.Text)
36.          + ';' + #13#10);
37.        call1.Caption := 'End';
38.      end;
39.    if LT1.Caption = '10' then
40.      begin
41.        memo2.Clear;
42.        mem1.Text := mem1.Text + #13#10 + 'Ending Call';
```

```

43. Comport1.WriteString('ATH'+ #13#10);
44. call1.Caption := 'Call';
45. end;
46. if LT1.Caption = '11' then
47. begin
48. memo2.Clear;
49. memo1.Text := memo1.Text + #13#10 + 'Processing
50. SMS';
51. Comport1.WriteString('AT+CMGS="' +
52. string(form2.utujuan2.text) + '"'+ #13#10);
53. Comport1.WriteString(form2.usms.Text);
54. Comport1.WriteString(#26);
55. end;
56. if LT1.Caption = '16' then
57. begin
58. memo2.Clear;
59. memo1.Text := memo1.Text + #13#10 + 'Processing
60. Registration';
61. Comport1.WriteString('AT+CMGS="'
62. +string(form2.utujuan1.Text) + '"'+ #13#10);
63. Comport1.WriteString(form2.ureg.Text + #26);
64. end;
65. if LT1.Caption = '21' then
66. begin
67. memo2.Clear;
68. memo1.Text := memo1.Text + #13#10 + 'Checking
69. Activation Status';
70. Comport1.WriteString('AT+CUSD=1,' +
71. string(form2.uussd.Text) + #13#10);
72. cek1.Caption := 'Cancel';
73. end;
74. end;

```

**Gambar 4.32 Implementasi Algoritma Connect**  
**Sumber : [Implementasi]**

Penjelasan implementasi algoritma *connect* dalam Gambar 4.33 yaitu:

1. Baris 3-5 merupakan proses untuk merubah warna *font* pada *field* nomor, masa aktif dan status menjadi warna hitam.
2. Baris 10-12 merupakan proses untuk mengatur *modem* menjadi *mode text*.
3. Baris 13-15 merupakan proses untuk mengatur media penyimpanan SMS pada *simcard*.
4. Baris 16-18 merupakan proses untuk mengatur karakter menjadi *GSM*.

5. Baris 20-22 merupakan proses untuk register ke jaringan.
6. Baris 24-26 merupakan proses untuk *unlock simcard fixed dial*.
7. Baris 30-38 merupakan proses untuk telepon ke nomor tujuan yang telah disimpan di *database* yang dijalankan pada detik ke 2.
8. Baris 39-45 merupakan proses untuk menutup telepon yang dijalankan pada detik ke 10.
9. Baris 46-55 merupakan proses untuk melakukan proses SMS yang mengambil data SMS dan nomor tujuan SMS di *database* yang dijalankan pada detik ke 11.
10. Baris 56-64 merupakan proses untuk melakukan proses *registrasi* yang mengambil data *registrasi* dan nomor tujuan *registrasi* di *database* yang dijalankan pada detik ke 16.
11. Baris 21-73 merupakan proses untuk mengecek status *simcard* yang dijalankan pada detik ke 21.

#### 4.2.4.2 Implementasi Algoritma Save

Operasi pada proses *save* yang ada di halaman utama bertujuan untuk melakukan penyimpanan data *simcard* yang telah melakukan aktivasi. Gambar 4.34 merupakan *procedure* dari *save* yang menyimpan data nomor, masa aktif dan status *simcard*.

```

1. procedure TForm1.saveClick(Sender: TObject);
2. begin
3.   if N1.Text <> '' then
4.     begin
5.       q1.SQL.Clear;
6.       q1.SQL.Add('select count(*) as n from dataact where
7.         msisdn like"' + string(N1.Text) + '"');
8.       q1.Open;
9.       if q1.FieldByName('n').AsString = '0' then
10.        begin
11.          q1.SQL.Clear;
12.          q1.SQL.Add('INSERT INTO dataact SET msisdn="' +
13.            string(N1.Text) + '" , masaaktif="' +
14.            string(A1.Text) + '" , waktuact=NOW() , statusact="'
15.            +
16.            string(S1.Text) + '"');
17.          q1.ExecSQL;

```

```
18.     end;
19.     q1.SQL.Clear;
20.     q1.SQL.Add('select count(*) as n from dataact where
21.         msisdn like"' + string(N1.Text) + "'');
22.     q1.Open;
23.     if q1.FieldByName('n').AsString <> '0' then
24.         begin
25.             q1.SQL.Clear;
26.             q1.SQL.Add('UPDATE dataact SET masaaktif="' +
27.                 string(A1.Text) + '" , statusact="' +
28.                 string(S1.Text) + '" where msisdn="' +
29.                 string(N1.Text) + "'');
30.             q1.ExecSQL;
31.             n1.Font.Color := clblue;
32.             if string(a1.Text)<>' ' then
33.                 a1.Font.Color := clblue;
34.             if string(s1.Text)<>' ' then
35.                 s1.Font.Color := clblue;
36.         end;
37.     end;
```

**Gambar 4.33 Implementasi Algoritma Save**  
**Sumber : [Implementasi]**

Penjelasan implementasi algoritma *save* dalam Gambar 4.34 yaitu:

1. Baris 5-8 merupakan proses untuk mengecek data pada *database* yang telah ditentukan.
2. Baris 11-17 merupakan proses untuk memasukkan data nomor, masa aktif, waktu aktivasi dan status aktivasi pada *database*.
3. Baris 25-30 merupakan proses untuk mengubah data nomor, masa aktif, waktu aktivasi, dan status aktivasi pada *database*.
4. Baris 31-35 merupakan proses untuk mengubah warna *font* nomor, masa aktif dan status aktivasi pada *field* nomor, *field* masa aktif dan *field* status aktivasi menjadi biru.

#### 4.2.4.3 Implementasi Algoritma *Setting*

Operasi pada proses *setting* yang ada di halaman *setting* bertujuan untuk melakukan pengaturan dengan mengisi data-data yang dibutuhkan untuk melakukan aktivasi. Gambar 4.35 merupakan *procedure* untuk menyimpan data-data yang telah diisi pada *database*.

```
1. procedure TForm2.Button1Click(Sender: TObject);
2. begin
3.     qsave.SQL.Clear;
4.     qsave.SQL.Add('update sett_app set isi_setting="' +
5.         string(ureg.Text) + " where
6.             nama_setting="Registrasi'");
7.     qsave.ExecSQL;
8.     qsave.Close;
9.     qsave.SQL.Clear;
10.    qsave.SQL.Add('update sett_app set isi_setting="' +
11.        string(usms.Text) + " where nama_setting="SMS'");
12.    qsave.ExecSQL;
13.    qsave.Close;
14.    qsave.SQL.Clear;
15.    qsave.SQL.Add('update sett_app set isi_setting="' +
16.        string(umaster.Text) + " where nama_setting="Master
17.            Transfer'");
18.    qsave.ExecSQL;
19.    qsave.Close;
20.    qsave.SQL.Clear;
21.    qsave.SQL.Add('update sett_app set isi_setting="' +
22.        string(ucall.Text) + " where nama_setting="Test
23.            Call'");
24.    qsave.ExecSQL;
25.    qsave.Close;
26.    qsave.SQL.Clear;
27.    qsave.SQL.Add('update sett_app set tujuan="' +
28.        string(utujuan1.Text) + " where
29.            nama_setting="Registrasi'");
30.    qsave.ExecSQL;
31.    qsave.Close;
32.    qsave.SQL.Clear;
33.    qsave.SQL.Add('update sett_app set tujuan="' +
34.        string(utujuan2.Text) + " where nama_setting="SMS'");
35.    qsave.ExecSQL;
36.    qsave.Close;
37.    qsave.SQL.Clear;
38.    qsave.SQL.Add('update sett_app set isi_setting="' +
39.        string(uussd.Text) + " where nama_setting="USSD'");
40.    qsave.ExecSQL;
41.    qsave.Close;
42.    qsave.SQL.Clear;
43.    qsave.SQL.Add('update sett_app set isi_setting="' +
44.        string(usyntax.Text) + " where
45.            nama_setting="syntax'");
46.    qsave.ExecSQL;
```

47.	qsave.Close;
48.	Form2.Hide;
49.	end;

**Gambar 4.34 Implementasi Algoritma *Setting***  
**Sumber : [Implementasi]**

Penjelasan implementasi algoritma *setting* dalam Gambar 4.35 yaitu:

1. Baris 3-8 merupakan proses untuk menyimpan data yang diperlukan untuk *registrasi* pada *database*.
2. Baris 9-13 merupakan proses untuk menyimpan data yang diperlukan untuk SMS pada *database*.
3. Baris 14-19 merupakan proses untuk menyimpan data nomor yang digunakan untuk nomor tujuan *transfer* pulsa pada *database*.
4. Baris 20-25 merupakan proses untuk menyimpan data nomor yang digunakan untuk nomor tujuan telepon pada *database*.
5. Baris 26-31 merupakan proses untuk menyimpan data nomor yang digunakan untuk nomor tujuan *registrasi* pada *database*.
6. Baris 32-36 merupakan proses untuk menyimpan data nomor yang digunakan untuk nomor tujuan SMS pada *database*.
7. Baris 37-41 merupakan proses untuk menyimpan data nomor yang digunakan untuk pengecekan status *simcard* pada *database*.
8. Baris 42-47 merupakan proses untuk menyimpan data *syntax* untuk *transfer* pulsa pada *database*.
9. Baris 48 merupakan proses untuk menutup *form setting*.

#### 4.2.4.4 Implementasi Algoritma Data

Operasi pada proses data yang ada di halaman data bertujuan untuk menampilkan data yang telah tersimpan pada *database*. Gambar 4.36 merupakan *procedure* untuk melihat data yang telah tersimpan di *database*.

1.	procedure TForm3.Button1Click(Sender: TObject);
2.	begin
3.	qdata.Close;
4.	qdata.SQL.Clear;
5.	qdata.SQL.Add('select MSISDN, MasaAktif, WaktuAct,
6.	StatusAct from dataact where date(WaktuAct)>=''
7.	+ FormatDateTime('yyyy-mm-dd', awal.Date) +' and

```

8.         date(WaktuAct) <= "
9.     + FormatDateTime('yyyy-mm-dd', akhir.Date) + "'";
10.    qdata.Open;
11.    qjumlah.Close;
12.    qjumlah.SQL.Clear;
13.    qjumlah.SQL.Add('select count(*) as jumlah from dataact
14.        where date(WaktuAct) <= "
15.    + FormatDateTime('yyyy-mm-dd', akhir.Date) + "'');
16.    qjumlah.Open;
17.    jmldata.Caption                               :=
18.        qjumlah.FieldName('jumlah').AsString;
19.    end;

```

**Gambar 4.35 Implementasi Algoritma Data**  
**Sumber : [Implementasi]**

Penjelasan implementasi algoritma data dalam Gambar 4.36 yaitu:

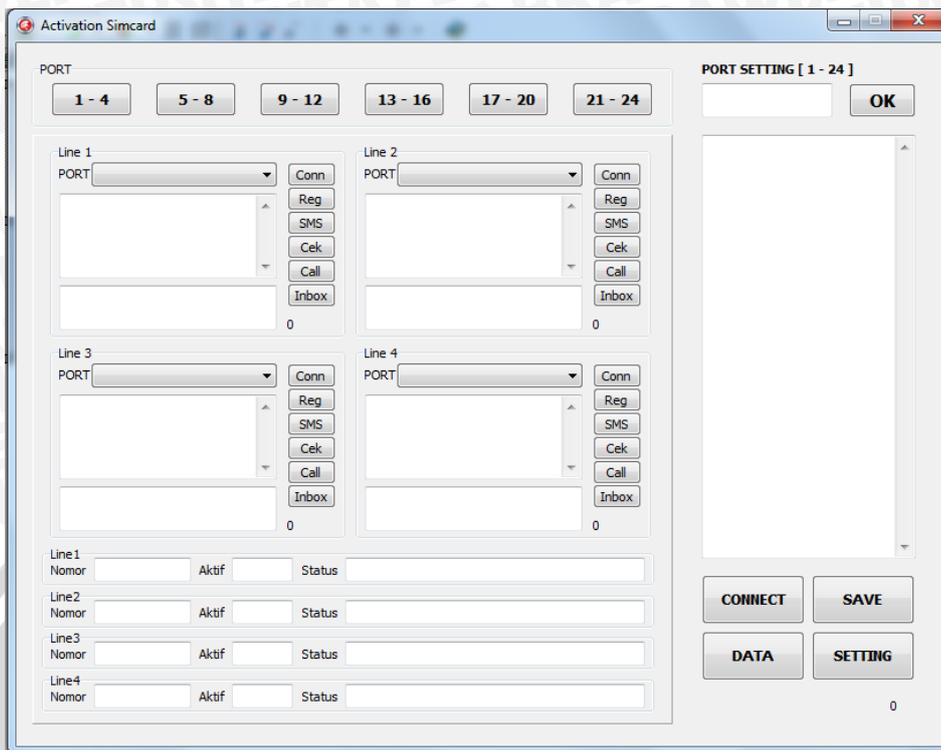
1. Baris 3-10 merupakan proses untuk menampilkan data nomor, masa aktif, waktu aktivasi, dan status aktivasi pada *database* sesuai tanggal yang telah ditentukan.
2. Baris 11-19 merupakan proses untuk menjumlah data yang ditampilkan.

#### 4.2.5 Implementasi Antarmuka

Antarmuka Aplikasi *Activation Simcard* menggunakan *modem pool* sebagai alat aktivasi digunakan oleh pengguna untuk berinteraksi dengan sistem perangkat lunak. Antarmuka perangkat lunak ini dibagi menjadi dua, yaitu antarmuka halaman utama dan antarmuka halaman *setting*.

##### 4.2.5.1 Implementasi Antarmuka Halaman Utama

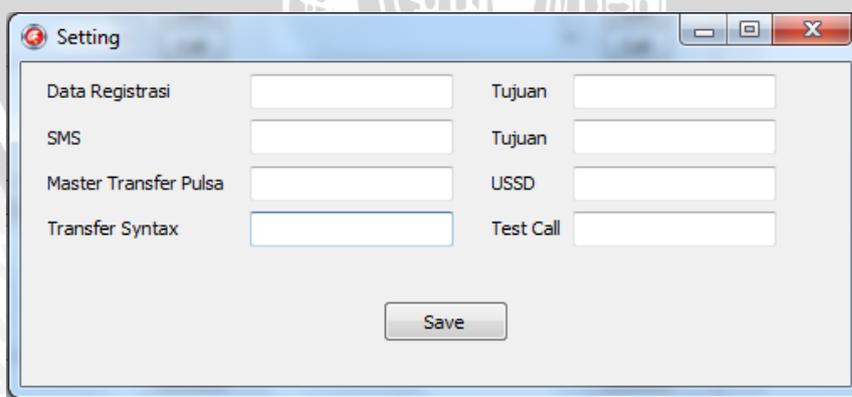
Halaman utama menampilkan *form* yang berisi *groupbox-groupbox* setiap *port* dan data aktivasi. Gambar 4.37 akan menunjukkan antarmuka halaman utama yang mengacu pada perancangan antarmuka halaman masuk Sub Bab 4.1.2.3 bagian a.



**Gambar 4.36 Implementasi Antarmuka Halaman Utama**  
 Sumber : [Implementasi]

#### 4.2.5.2 Implementasi Antarmuka Halaman *Setting*

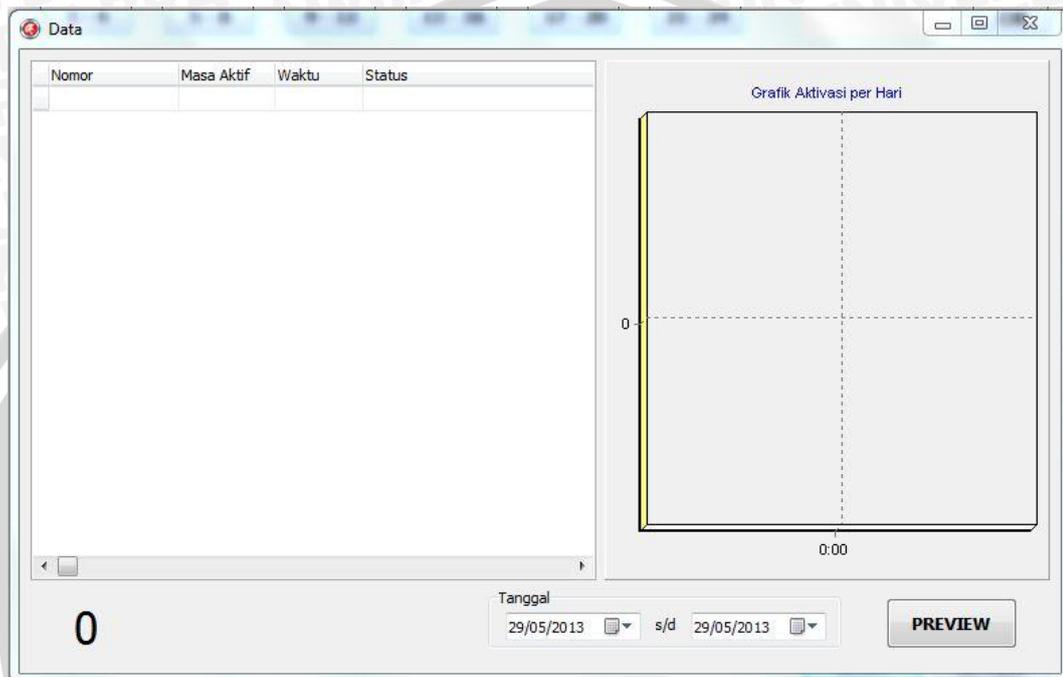
Halaman *setting* menampilkan *form* yang berisi *field* yang akan diisi dengan data-data yang diperlukan untuk aktivasi. Gambar 4.38 akan menunjukkan antarmuka halaman *setting* yang mengacu pada perancangan antarmuka halaman masuk Sub Bab 4.1.2.3 bagian b.



**Gambar 4.37 Implementasi Antarmuka Halaman *Setting***  
 Sumber : [Implementasi]

#### 4.2.5.3 Implementasi Antarmuka Halaman Data

Halaman data menampilkan *form* yang berisi *field* yang akan menampilkan data yang telah tersimpan di *database*. Gambar 4.39 akan menunjukkan antarmuka halaman data yang mengacu pada perancangan antarmuka halaman masuk Sub Bab 4.1.2.3 bagian c.



Gambar 4.38 Implementasi Antarmuka Halaman Data  
Sumber : [Implementasi]



## BAB V

### PENGUJIAN DAN ANALISIS

Bab ini membahas mengenai tahapan pengujian dan analisis perangkat lunak *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi. Proses pengujian dilakukan melalui dua macampengujian yaitu pengujian validasi dan pengujian performa. Pengujian validasi digunakan untuk menguji kebenarantalannya kebutuhan fungsional perangkat lunak *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi. Pengujian performa digunakan untuk menguji kecepatan proses sistem perangkat lunak. Proses pengujian yang dilakukan melalui duamacam pengujian yaitu pengujian validasi dan pengujian performa.

Proses analisis bertujuan untuk mendapatkan kesimpulan dari hasil pengujian perangkat lunak *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasiyang telah dilakukan. 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 validasi dan pengujian performa.

#### 5.1 Pengujian

Proses pengujian dilakukan melalui dua tahapan (strategi) yaitu pengujian pengujian validasi dan pengujian performa aplikasi perangkat lunak *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi.

##### 5.1.1 Pengujian Validasi

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 validasi berfungsi untuk menemukan kesesuaian antara kerja sistem dengan daftar kebutuhan yang telah dirancang sebelumnya. Pada skripsi ini dilakukan pengujian validasi terhadap perangkat lunak *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi.

### 5.1.1.1 Kasus dan Hasil Pengujian Validasi

#### a. Kasus Uji Melakukan *Setting*

**Tabel 5.1 Kasus Uji Melakukan *Setting***

Nama Kasus Uji	Kasus Uji Melakukan <i>Setting</i>
Objek Uji	Kebutuhan Fungsional (SRS_001)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melakukan <i>setting</i> dan menyimpan data <i>setting</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. <i>User</i> menekan tombol <i>Setting</i>.</li> <li>3. <i>User</i> mengisi data pada <i>field</i> yang telah disediakan.</li> <li>4. <i>User</i> menekan tombol <i>save</i>.</li> </ol>
Hasil yang Diharapkan	Aplikasi dapat melakukan <i>setting</i> dan menyimpan data <i>setting</i> .
Hasil yang Didapatkan	Aplikasi dapat melakukan <i>setting</i> dan menyimpan data <i>setting</i> .
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

#### b. Kasus Uji Melakukan *Connect*

**Tabel 5.2 Kasus Uji Melakukan *Connect***

Nama Kasus Uji	Kasus Uji Melakukan <i>Connect</i>
Objek Uji	Kebutuhan Fungsional (SRS_002)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melakukan semua proses aktivasi untuk semua simcard tanpa harus satu persatu.
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. <i>User</i> menekan tombol <i>Connect</i>.</li> </ol>
Hasil yang Diharapkan	Aplikasi dapat melakukan semua proses aktivasi untuk semua simcard tanpa harus satu persatu.
Hasil yang Didapatkan	Aplikasi dapat melakukan semua proses aktivasi untuk semua

	simcard tanpa harus satu persatu.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

c. Kasus Uji Melakukan *Registrasi*

**Tabel 5.3 Kasus Uji Melakukan *Registrasi***

Nama Kasus Uji	Kasus Uji Melakukan <i>Registrasi</i>
Objek Uji	Kebutuhan Fungsional (SRS_003)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk mengirim sms menggunakan <i>modem pool</i> untuk melakukan <i>registrasi</i> .
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menekan tombol Reg.
Hasil yang Diharapkan	Aplikasi dapat mengirim sms menggunakan <i>modem pool</i> untuk melakukan <i>registrasi</i> dengan data registrasi yang telah tersimpan.
Hasil yang Didapatkan	Aplikasi dapat mengirim sms menggunakan <i>modem pool</i> untuk melakukan <i>registrasi</i> dengan data registrasi yang telah tersimpan.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

d. Kasus Uji Melakukan SMS

**Tabel 5.4 Kasus Uji Melakukan SMS**

Nama Kasus Uji	Kasus Uji Melakukan SMS
Objek Uji	Kebutuhan Fungsional (SRS_004)
Tujuan Pengujian	Pengujian dilakukan untuk mengirim sms menggunakan <i>modem pool</i> untuk melakukan sms atau bisa dimanfaatkan untuk claim.
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menekan tombol SMS.
Hasil yang Diharapkan	Aplikasi dapat mengirim sms menggunakan <i>modem pool</i> untuk melakukan sms atau bisa dimanfaatkan untuk claim.

Hasil yang Didapatkan	Aplikasi dapat mengirim sms menggunakan <i>modem pool</i> untuk melakukan sms atau bisa dimanfaatkan untuk claim.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

e. Kasus Uji Melakukan Telepon

**Tabel 5.5 Kasus Uji Melakukan Telepon**

Nama Kasus Uji	Kasus Uji Melakukan Telepon
Objek Uji	Kebutuhan Fungsional (SRS_005)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melakukan telepon atau dial up menggunakan <i>modem pool</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. User menekan tombol <i>Call</i>.</li> </ol>
Hasil yang Diharapkan	Aplikasi dapat melakukan telepon atau dial up menggunakan <i>modem pool</i> ke nomor yang telah tersimpan.
Hasil yang Didapatkan	Aplikasi dapat melakukan telepon atau dial up menggunakan <i>modem pool</i> ke nomor yang telah tersimpan.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

f. Kasus Uji Melakukan *Transfer* Pulsa

**Tabel 5.6 Kasus uji Melakukan *Transfer* Pulsa**

Nama Kasus Uji	Kasus Uji Melakukan <i>Transfer</i> Pulsa
Objek Uji	Kebutuhan Fungsional (SRS_006)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melakukan <i>transfer</i> pulsa menggunakan <i>modem pool</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. User menekan tombol <i>Cek</i>.</li> </ol>

Hasil yang Diharapkan	Aplikasi dapat melakukan <i>transfer</i> pulsa apabila jumlah pulsa lebih dari 6000 dan akan langsung di <i>transfer</i> ke nomor yang telah tersimpan.
Hasil yang Didapatkan	Aplikasi dapat melakukan melakukan <i>transfer</i> pulsa apabila jumlah pulsa lebih dari 6000 dan akan langsung di <i>transfer</i> ke nomor yang telah tersimpan.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

g. Kasus Uji Melakukan Pengecekan

**Tabel 5.7 Kasus uji Melakukan Pengecekan**

Nama Kasus Uji	Kasus Uji Melakukan Pengecekan
Objek Uji	Kebutuhan Fungsional (SRS_007)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk melakukan pengecekan nomor dan masa aktif menggunakan <i>modem pool</i> .
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menekan tombol <i>Cek</i> .
Hasil yang Diharapkan	Aplikasi dapat melakukan pengecekan nomor dan masa aktif menggunakan <i>modem pool</i> .
Hasil yang Didapatkan	Aplikasi dapat melakukan pengecekan nomor dan masa aktif menggunakan <i>modem pool</i> .
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

h. Kasus Uji Mengecek *Inbox*

**Tabel 5.8 Kasus Uji Mengecek *Inbox***

Nama Kasus Uji	Kasus Uji Mengecek <i>Inbox</i>
Objek Uji	Kebutuhan Fungsional (SRS_008)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk mengecek <i>inbox</i> bila ada

	sms masuk ke simcard telah mendapatkan laporan <i>registrasi</i> berhasil.
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> menekan tombol <i>Inbox</i> .
Hasil yang Diharapkan	Aplikasi dapat melakukan semua sms masuk yang telah tersimpan di <i>Inbox simcard</i> dan dapat menampilkan sms laporan registrasi berhasil.
Hasil yang Didapatkan	Aplikasi dapat melakukan semua sms masuk yang telah tersimpan di <i>Inbox simcard</i> dan dapat menampilkan sms laporan registrasi berhasil.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

i. Kasus Uji Melakukan Penyimpanan Data

**Tabel 5.9 Kasus Uji Melakukan Penyimpanan Data**

Nama Kasus Uji	Kasus Uji Melakukan Penyimpanan Data
Objek Uji	Kebutuhan Fungsional (SRS_009)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menyimpan data nomor, masa aktif dan status <i>simcard</i> yang telah diaktifkan ke basis data.
Prosedur Uji	1. Sistem dijalankan saat program dieksekusi. 2. <i>User</i> melakukan aktivasi <i>simcard</i> . 3. <i>User</i> menekan tombol <i>Save</i> .
Hasil yang Diharapkan	Aplikasi dapat menyimpan data nomor, masa aktif dan status <i>simcard</i> yang telah diaktifkan ke basis data.
Hasil yang Didapatkan	Aplikasi dapat menyimpan data nomor, masa aktif dan status <i>simcard</i> yang telah diaktifkan ke basis data.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

## j. Kasus UjiMelihat Data

Tabel 5.10 Kasus Uji Melihat Data

Nama Kasus Uji	Kasus UjiMelihat Data
Objek Uji	Kebutuhan Fungsional (SRS_010)
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional untuk menampilkan data <i>simcard</i> yang telah di aktivasi dan mengetahui jumlah data yang telah di aktivasi sesuai dengan rentang tanggal yang telah ditentukan.
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan saat program dieksekusi.</li> <li>2. <i>User</i> menekan tombol Data.</li> <li>3. <i>User</i> menentukan rentang tanggal data yang akan dilihat.</li> <li>4. <i>User</i>menekan tombol <i>Preview</i>.</li> </ol>
Hasil yang Diharapkan	Aplikasi dapat menampilkan data <i>simcard</i> yang telah di aktivasi dan dapatmenjumlah data yang telah di aktivasi sesuai dengan rentang tanggal yang telah ditentukan.
Hasil yang Didapatkan	Aplikasi dapat menampilkan data <i>simcard</i> yang telah di aktivasi dan dapatmenjumlah data yang telah di aktivasi sesuai dengan rentang tanggal yang telah ditentukan.
Status Validitas	Valid

Sumber : [Pengujian dan Analisis]

## 5.1.2 Pengujian Performa

Pengujian performa digunakan untuk mengetahui apakah sistem yang dibangun dapat melakukan proses aktivasi secara cepat, sehingga dapat memberikan kepastian kepada agen dengan cepat pula. Selain itu pengujian ini dilakukan dengan cara bertahap dimana dilakukan pengujian pada satu *simcard* terlebih dahulu lalu melakukan pengujian secara *multi*. Pada pengujian satu *simcard* akan dilakukan dengan tujuan untuk mencari rentang waktu terbaik pada masing-masing proses yang terjadi. Sedangkan untuk pengujian secara *multi* dilakukan secara bertahap 4 *simcard*, 8 *simcard*, dan 16 *simcard* 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 *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi.

**Tabel 5.11 Hasil Pengujian Performa Satu Simcard**

Jeda Waktu (s)	Call	SMS	Registrasi	Cek
1	Error	Error	Error	Error
2	Error	Error	Error	Error
3	Error	Error	Error	Error
4	Error	Ok	Error	Error
5	Error	Ok	Ok	Ok
6	Error	Ok	Ok	Ok
7	Error	Ok	Ok	Ok
8	Ok	Ok	Ok	Ok

Sumber : [Pengujian dan Analisis]

Dari tabel 5.11 pengujian dilakukan secara bertahap. Untuk pengujian pertama dilakukan pengujian dengan selisih waktu setiap proses 1 detik dan didapatkan semua proses terjadi *error*. Semua proses terjadi *error* karena waktu setiap proses membutuhkan waktu yang lebih lama. Selain itu proses bisa terjadi *error* ketika satu proses tersebut belum berhasil dan proses selanjutnya dijalankan. Pada pengujian kedua dan ketiga juga memiliki hasil yang sama. Pengujian keempat proses SMS berhasil tetapi proses yang lain terjadi *error*. Pada pengujian kelima dengan selisih waktu 5 detik, proses SMS, *registrasi*, dan cek berhasil, sedangkan untuk proses *call* masih terjadi *error*. Proses *call* dapat berhasil ketika proses *call* berjalan selama 8 detik. Dari pengujian tersebut dapat disimpulkan bahwa setiap proses yang dilakukan memiliki selisih waktu yang berbeda. Dari beberapa kali pergantian selisih waktu yang diberikan, proses *call* membutuhkan waktu yang paling lama.

Maka untuk proses pengujian satu simcard setiap proses memiliki waktu terbaik sebagai berikut :

1. *Call* = 8 detik
2. SMS = 5 detik
3. *Registrasi* = 5 detik
4. Cek = 5 Detik

Setelah melakukan pengujian dengan satu *simcard* dan mendapatkan waktu terbaik selanjutnya dilakukan pengujian secara multi dimana akan

dilakukan bertahap dari 4 *simcard*, 8 *simcard* dan 16 *simcard*. Pengujian secara multi ini bertujuan untuk mendapatkan performa aplikasi ketika melakukan aktivasi beberapa *simcard* dengan melihat persentase keberhasilan dan waktu tunggu yang menunjukkan lama proses.

**Tabel 5.12 Hasil pengujian performa 4 Simcard**

id	Call	SMS	Registrasi	Cek	Persentase	Waktu
1	Ok	Ok	Ok	Error	75%	76 detik
2	Ok	Ok	Ok	Ok	100%	83 detik
3	Ok	Ok	Ok	Ok	100%	70 detik
4	Error	Ok	Ok	Ok	75%	73 detik
Rata-rata					87,5%	75,5 detik

Sumber : [Pengujian dan Analisis]

Dari tabel 5.12 pengujian dengan 4 *simcard* terjadi error pada *simcard* 1 dan 4. Pada *simcard* 1 terjadi error pada proses cek sehingga didapatkan persentase keberhasilan 75% dari 4 proses yang dilakukan. Pada *simcard* ke 4 terjadi *error* pada proses *call* sehingga juga didapatkan persentase keberhasilan 75%. Untuk *simcard* ke 2 dan 3 memiliki persentase keberhasilan 100% karena semua proses berhasil. Proses yang terjadi *error* disebabkan karena jaringan dari *provider*, sehingga ketika melakukan proses tersebut membutuhkan waktu yang lebih lama agar tidak terjadi *error*. Pada pengujian ini dapat disimpulkan bahwa proses aktivasi untuk 4 *simcard* secara langsung memiliki rata-rata *persentase* keberhasilan 87,5% dan rata-rata waktu tunggu 75,5 detik.

**Tabel 5.13 Hasil pengujian performa 8 Simcard**

id	Call	SMS	Registrasi	Cek	Persentase	Waktu
1	Ok	Ok	Ok	Ok	100%	78 detik
2	Ok	Ok	Ok	Ok	100%	76 detik
3	Ok	Ok	Ok	Ok	100%	80 detik
4	Error	Ok	Ok	Ok	75%	77 detik
5	Ok	Ok	Ok	Ok	100%	80 detik
6	Ok	Ok	Error	Error	50%	76 detik
7	Ok	Ok	Ok	Ok	100%	79 detik
8	Ok	Ok	Ok	Ok	100%	78 detik
Rata-rata					90,625%	78 detik

Sumber : [Pengujian dan Analisis]

Dari tabel 5.13 pengujian dengan 8 *simcard* terjadi error pada *simcard* 4 dan 6. Pada *simcard* 4 terjadi error pada proses *call* sehingga didapatkan persentase keberhasilan 75% dari 4 proses yang dilakukan. Pada *simcard* ke 6

terjadi *error* pada proses *registrasi* dan cek sehingga didapatkan persentase keberhasilan 50%. Untuk *simcard* yang lain memiliki persentase keberhasilan 100% karena semua proses berhasil. Proses yang terjadi *error* disebabkan karena jaringan dari *provider*, sehingga ketika melakukan proses tersebut membutuhkan waktu yang lebih lama agar tidak terjadi *error*. Dari pengujian ini dapat disimpulkan bahwa proses aktivasi untuk 8 *simcard* secara langsung memiliki rata-rata *persentase* keberhasilan 90,625% dan rata-rata waktu tunggu 78 detik.

**Tabel 5.14 Hasil pengujian performa 16 Simcard**

id	Call	SMS	Registrasi	Cek	Persentase	Waktu
1	Ok	Ok	Ok	Ok	100%	79 detik
2	Ok	Ok	Ok	Ok	100%	77 detik
3	Ok	Ok	Ok	Ok	100%	82 detik
4	Ok	Ok	Ok	Ok	100%	80 detik
5	Ok	Ok	Error	Error	50%	79 detik
6	Ok	Ok	Ok	Ok	100%	78 detik
7	Ok	Ok	Ok	Ok	100%	76 detik
8	Ok	Ok	Ok	Error	75%	77 detik
9	Ok	Ok	Ok	Ok	100%	79 detik
10	Ok	Ok	Ok	Ok	100%	83 detik
11	Ok	Ok	Ok	Ok	100%	81 detik
12	Ok	Ok	Ok	Ok	100%	78 detik
13	Error	Ok	Ok	Ok	75%	77 detik
14	Ok	Ok	Ok	Ok	100%	79 detik
15	Ok	Ok	Ok	Ok	100%	82 detik
16	Ok	Ok	Ok	Ok	100%	78 detik
Rata-rata					93,75%	79,06 detik

**Sumber : [Pengujian dan Analisis]**

Dari tabel 5.14 pengujian dengan 16 *simcard* terjadi *error* pada *simcard* 5, 8 dan 13. Pada *simcard* 5 terjadi *error* pada proses *registrasi* dan cek sehingga didapatkan persentase keberhasilan 50%. Pada *simcard* ke 8 terjadi *error* pada proses cek sehingga juga didapatkan persentase keberhasilan 75%. Pada *simcard* ke 13 terjadi *error* pada proses *call* sehingga juga didapatkan persentase keberhasilan 75%. Untuk *simcard* yang lain memiliki persentase keberhasilan 100% karena semua proses berhasil. Proses yang terjadi *error* bisa disebabkan karena jaringan dari *provider*, sehingga ketika melakukan proses tersebut membutuhkan waktu yang lebih lama agar tidak terjadi *error*. Dari pengujian ini dapat disimpulkan bahwa proses aktivasi untuk 16 *simcard* secara langsung

memiliki rata-rata *persentase* keberhasilan 93,75% dan rata-rata waktu tunggu 79,06 detik.

## 5.2 Analisis

Proses analisis bertujuan untuk mendapatkan kesimpulan dari hasil pengujian perangkat lunak *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi yang telah dilakukan. 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 validasi dan analisis hasil pengujian performa.

### 5.2.1 Analisis Hasil Pengujian Validasi

Proses analisis terhadap hasil pengujian validasi dilakukan dengan melihat konformitas antara hasil kinerja sistem dengan daftar kebutuhan. Berdasarkan hasil pengujian validasi dapat disimpulkan bahwa implementasi dan fungsionalitas perangkat lunak *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi telah memenuhi kebutuhan yang telah dijabarkan pada tahap analisis kebutuhan.

### 5.2.2 Analisis Hasil Pengujian Performa

Proses analisis terhadap hasil pengujian performa aktivasi *simcard* secara multi. Berdasarkan hal tersebut maka dapat diambil kesimpulan sebagai berikut :

- a. Jaringan *provider* dan tempat dapat mempengaruhi *persentase* keberhasilan dan waktu aktivasi.
- b. Setiap proses memiliki rentang waktu yang berbeda agar proses tidak terjadi *error*.
- c. Proses yang *error* dikarenakan proses tersebut membutuhkan waktu yang lebih lama, sehingga ketika proses tersebut masih berjalan dan belum berhasil dan sudah ke proses berikutnya maka proses tersebut akan mengalami *error* karena belum terselesaikan. Selain itu proses bisa terjadi *error* juga dikarenakan jaringan dari *provider* tersebut.
- d. Rentang waktu yang terbaik untuk setiap proses setelah dilakukan pengujian adalah sebagai berikut:

- *Call* = 8 detik

- *SMS* = 5 detik
  - *Registrasi* = 5 detik
  - *Cek* = 5 Detik
- e. Pengujian secara multi untuk aktivasi 4 *simcard* memiliki rata-rata persentase keberhasilan 87,5% dan rata-rata waktu tunggu 75,5 detik.
- f. Pengujian secara multi untuk aktivasi 8 *simcard* memiliki rata-rata persentase keberhasilan 90,625% dan rata-rata waktu tunggu 78 detik.
- g. Pengujian secara multi untuk aktivasi 16 *simcard* memiliki rata-rata persentase keberhasilan 93,75% dan rata-rata waktu tunggu 79,06 detik.
- h. Dari pengujian yang dilakukan apabila aktivasi dilakukan secara manual dengan jumlah 20000 *simcard* dan dilakukan oleh 4 orang membutuhkan waktu 2666,67 jam, karena proses aktivasi satu *simcard* membutuhkan waktu 8 menit sehingga dalam satu jam dapat melakukan aktivasi rata-rata 7 atau 8 *simcard*.
- i. Dari hasil pengujian yang dilakukan untuk aktivasi 8 *simcard* memiliki rata-rata waktu 78 detik dan untuk proses secara keseluruhan membutuhkan waktu 3 menit. Sehingga untuk aktivasi 20000 *simcard* membutuhkan 125 jam, karena proses aktivasi *simcard* dalam satu jam dapat melakukan aktivasi 160 *simcard*.

## BAB VI PENUTUP

Berdasarkan perancangan, implementasi dan hasil pengujian dari sistem, maka didapatkan kesimpulan sebagai berikut:

### 6.1 Kesimpulan

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

1. Aplikasi *Activation Simcard* dengan menggunakan *modem pool* sebagai alat aktivasi dapat melakukan *call*, *SMS*, *registrasi*, *transfer pulsa* dan menyimpan data aktivasi.
2. Jaringan *provider* dan tempat dapat mempengaruhi *persentase* keberhasilan dan waktu tunggu aktivasi.
3. Setiap proses memiliki rentang waktu yang berbeda untuk memperoleh hasil yang diinginkan. Rentang waktu terbaik yang diperoleh untuk proses *call* 8 detik dan untuk proses *SMS*, *registrasi*, dan cek 5 detik.
4. Pengujian secara multi untuk aktivasi 4 *simcard*, 8 *simcard*, dan 16 *simcard* memiliki rata-rata *persentase* keberhasilan 90,625% dan rata-rata waktu tunggu 77,52 detik.
5. Hasil pengujian dengan menggunakan aplikasi ini untuk aktivasi 20000 *simcard* membutuhkan waktu 125 jam, karena dalam satu jam dalam melakukan aktivasi 160 *simcard*.

### 6.2 Saran

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

1. Aplikasi disarankan untuk digunakan pada tempat dan waktu yang tidak padat jaringannya karena mempengaruhi *persentase* keberhasilan dan waktu tunggu.
2. Untuk pengembangan lebih lanjut, disarankan untuk membuat aplikasi yang dapat melakukan aktivasi untuk beberapa *provider* karena dapat memberikan solusi untuk aktivasi *simcard provider* lain apabila ada event dari *provider* lain.

## DAFTAR PUSTAKA

- [DJA-05] Djalil, Sofyan A. 2005. *Registrasi Pelanggan Jasa Telekomunikasi*. Jakarta : Menteri Komunikasi Dan Informatika.
- [KAD-01] Kadir, Abdul. 2001. *Dasar Pemrograman Delphi 5.0 Jilid 1*. Yogyakarta : Andi.
- [KAV-10] Katankar, Veena. 2010. *Short Message Service using SMS Gateway*. Vol 2. Amravaty, Amravaty University.
- [MSK-09] Muhammad Salem, dan Kyung-Goo Doh. 2009. *Generic Information System using SMS Gateway*. Ansan, Hanyang University.
- [PRE-01] Pressman, R. 2001. *Software Engineering : A Practitioner's Approach, Fifth Edition*. McGraw Hill.
- [PUR-10] Purnamasari, Chindie. 2010. *Implementasi SMS Gateway dalam Solusi Penyediaan Laporan Peserta Didik Kursus*. Bandung, Politeknik Telkom Bandung.
- [SOM-11] Sommerville, Ian. 2011. *Software Engineering, Ninth Edition*. Boston, Addison-Wesley.
- [TFK-08] Taupan Aryo Dwitomo, Firmansyah Apyradhi, dan Kristian Nalindra Parlingoman. 2008. *Perancangan Aplikasi SMS Pencatatan KWH Meter dan Penghitungan Tagihan Untuk Pelanggan PT. PLN Lampung (PERSERO)*. Jakarta, Universitas Bina Nusantara.
- [WAV-02] Wavecom. 2002. *AT Commands Interface Guide*. Muse Platform.
- [WIY-11] Wiharto, Yudi. 2011. *Sistem Informasi Akademik Berbasis SMS Gateway*. Palembang, Politeknik PalComTech Palembang.