

**PROTOTYPE SISTEM RESERVASI TIKET BUS  
MENGUNAKAN PHP dan SMS GATEWAY  
(Studi Kasus di Terminal Arjosari Malang)**

**Konsentrasi : Rekayasa Perangkat Lunak**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer



Disusun Oleh :  
**Surya Wirawan**  
**NIM. 0710683007**

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN  
PROGRAM STUDI TEKNIK INFORMATIKA  
PROGRAM TEKNOLOGI INFORMASI DAN ILMU  
KOMPUTER**

**UNIVERSITAS BRAWIJAYA  
MALANG**

**2013**

## KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang selalu memberikan rahmat, hidayah dan cahaya-Nya sehingga penulis dapat menyelesaikan Proposal Skripsi dengan judul “ Prototipe Sistem Reservasi Tiket Bus Menggunakan PHP dan SMS GATEWAY (Studi Kasus di Terminal Arjosari Malang) “. Proposal skripsi ini diajukan sebagai salah satu syarat untuk mendapatkan gelar sarjana S-1 di Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.

Keberadaan skripsi ini tidak terlepas dari bimbingan dan bantuan dari berbagai pihak, untuk itu pada kesempatan ini penulis menyampaikan rasa terima kasih dan penghargaan sebesar- besarnya kepada:

1. Allah SWT
2. Ibu drh. Elisha Dewijanti, ayah Ir. Irwan Budiarto, adik Irma Kusumadewi dan keluarga yang tidak pernah bosan dan lupa untuk memberikan doa dan dorongan semangat hingga terselesainya proposal skripsi ini.
3. Bapak Arief Andy Soebroto, ST.,M.Kom., dan Ismiarta Aknuranda, ST., M.Sc., PhD., yang dengan sabar memberikan saran dan masukan perbaikan proposal skripsi ini.
4. Nurenza Deviningrum beserta keluarga serta teman- teman Program Studi Teknik Perangkat Lunak 2007 untuk bantuan dan semangatnya dan semua pihak yang tidak dapat disebutkan satu – persatu.

Malang, 19 Januari 2013

Penulis

## ABSTRAK

**Surya Wirawan. 2012. : Sistem Reservasi Tiket Bus Menggunakan PHP dan SMS GATEWAY (Studi Kasus Terminal Arjosari Malang). Skripsi Program Studi Teknik Informatika, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya.**

**Dosen Pembimbing : Arief Andy Soebroto, ST.,M.Kom dan Ismiarta Aknuranda, ST., M.Sc., PhD.**

Sistem Reservasi Tiket Bus adalah aplikasi yang dapat digunakan untuk membantu pemesanan tiket bus dan pembeli akan mendapatkan report melalui SMS Gateway. Analisis kebutuhan untuk reservasi tiket dibuat dengan melakukan studi lapang di Terminal Arjosari Malang. Sistem ini menggunakan SMS Gateway untuk mengirimkan laporan kepada pembeli setelah melakukan pemesanan tiket bus secara online. SMS Gateway pada aplikasi ini berfungsi sebagai penghubung yang melakukan *delay sms* antara External Short Message Entity (ESME) dan Short Message Service Center (SMSC). Sistem reservasi ini dikembangkan sebagai prototype dengan bahasa pemrograman PHP. Sistem ini diuji menggunakan pengujian validasi, pengujian performa, dan pengujian *usability*. Dari hasil pengujian validasi dapat disimpulkan bahwa semua proses dapat berjalan dengan baik sesuai dengan kasus ujinya. Waktu tunggu rata-rata *auto-reply* SMS dalam pengujian performa adalah 41.92 detik. Karakter layanan penyedia jasa SMS dan tempat pelanggan mengirim SMS berpengaruh terhadap waktu tunggu *auto-reply* SMS. Hasil persentase jawaban pengujian *usability* adalah 67,7%, menunjukkan bahwa sistem ini cukup dapat digunakan dengan baik dengan beberapa keterbatasan.

**Kata kunci :** *SMS Gateway*, Tiket bus, Terminal Arjosari Malang.

## ABSTRACT

**Surya Wirawan. 2012. : Bus Ticket Reservation System Using PHP and SMS GATEWAY (Arjosari Malang Terminal Case Study). Advisor : Arief Andy Soebroto, ST.,M.Kom dan Ismiarta Aknuranda, ST., M.Sc., PhD.**

Bus Ticket Reservation System is an application that can be used to help booking bus tickets and the customer will get a report via SMS Gateway. The requirement analysis of tickets reservation system is made by conducting field studies in Arjosari Terminal Malang. This system uses SMS Gateway to send the report to the customer after they book the bus tickets online. SMS Gateway on this application serves as a liaison which delays sms between External Short Message entity (ESME) and Short Message Service Center (SMSC).

This reservation system was developed as a prototype using PHP programming language. The system was tested using validation, performance, and usability testing. From the result of the validation testing, it can be inferred that all processes goes well corresponding to the testing case. The waiting period average of the SMS auto-reply in performance testing is 41.92 seconds. The quality of SMS provider and the place where the customer sends the SMS affect the waiting period of the SMS auto-reply. The total percentage of usability testing is 67.7%, showing that this system can be used well enough with some limitations.

**Keywords :** *SMS Gateway, bus ticket, Arjosari Terminal Malang.*

## DAFTAR ISI

<b>KATA PENGANTAR</b> .....	<b>i</b>
<b>ABSTRAK</b> .....	<b>ii</b>
<b>ABTRACT</b> .....	<b>iii</b>
<b>DAFTAR ISI</b> .....	<b>iv</b>
<b>DAFTAR TABEL</b> .....	<b>viii</b>
<b>DAFTAR GAMBAR</b> .....	<b>x</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>xii</b>
<b>BAB I</b>	
<b>PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian .....	3
1.6 Sistematika Penulisan.....	3
<b>BAB II</b>	
<b>DASAR TEORI</b> .....	<b>5</b>
2.1 Rekeyasa Perangkat Lunak .....	5
2.2 <i>Unified Modelling Language</i> .....	5
2.2.1 <i>Use Case Diagram</i> .....	7
2.2.2 <i>Data Flow Diagram</i> .....	9
2.2.3 <i>Sequence Diagram</i> .....	10
2.3 Basis Data .....	12
2.3.1 <i>Database Management System</i> .....	13
2.4 <i>Structured Query Language</i> .....	13
2.5 <i>Entity Relationship Diagram</i> .....	15
2.6 PHP .....	16
2.7 SMS GATEWAY .....	16
2.8 GAMMU .....	17

2.8.1 Instalasi Dan Konfigurasi GAMMU .....	17
2.9 Pengujian Perangkat Lunak .....	19
2.9.1 Teknik Pengujian .....	20
2.9.1.1 <i>Black-Box Testing</i> .....	20
2.9.2 Strategi Pengujian .....	20
2.9.2.1 Pengujian Validasi .....	21
2.9.2.2 Pengujian Performa .....	21
2.9.2.3 Pengujian <i>Usability</i> .....	22

### **BAB III**

#### **METODOLOGI PENELITIAN ..... 23**

3.1 Metode Penelitian .....	23
3.1.1 Studi Literatur .....	23
3.1.2 Analisa Kebutuhan .....	23
3.1.3 Perancangan Sistem .....	24
3.1.4 Implementasi .....	25
3.1.5 Pengujian Dan Analisis .....	25
3.1.6 Pengambilan Kesimpulan .....	25

### **BAB IV**

#### **PERANCANGAN ..... 26**

4.1 Analisis Kebutuhan .....	27
4.1.1 Gambaran Umum Perangkat Lunak .....	27
4.1.2 Identifikasi Aktor .....	30
4.1.3 Analisis Data .....	30
4.1.4 Analisis <i>Use Case</i> .....	31
4.1.4.1 Diagram <i>Use Case</i> Penumpang .....	32
4.1.4.2 Diagram <i>Use Case</i> Aplikasi Admin .....	37
4.2 Perancangan Data .....	45
4.2.1 Pemodelan Dengan ERD .....	45
4.2.2 Pemodelan Dengan DFD .....	48
4.3 Perancangan Antarmuka .....	54
4.3.1 Perancangan Alur Kerja Penumpang .....	54
4.3.2 Perancangan Antarmuka Penumpang .....	57

4.3.3 Perancangan Alur Kerja Admin .....	60
4.3.4 Perancangan Antarmuka Admin .....	68
<b>BAB V</b>	
<b>IMPLEMENTASI .....</b>	<b>74</b>
5.1 Spesifikasi Platform Pengembangan .....	74
5.1.1 Spesifikasi Lingkungan Perangkat Keras .....	75
5.1.2 Spesifikasi Lingkungan Perangkat Lunak .....	75
5.2 Batasan-Batasan Implementasi .....	75
5.3 Implementasi Basis Data .....	76
5.4 Implementasi <i>Use Case</i> dan Alur Kerja .....	78
5.4.1 Implementasi Proses <i>Login</i> .....	78
5.4.2 Implementasi Proses Mengelola Data Pengemudi .....	80
5.4.3 Implementasi Proses Mengelola Data Bus .....	82
5.4.4 Implementasi Proses Mengelola Data Jadwal Bus .....	85
5.4.5 Implementasi Proses Mengelola Data Pemesanan .....	87
5.5 Implementasi Antarmuka Aplikasi .....	88
5.5.1 Implementasi Antarmuka Admin .....	88
5.5.2 Implementasi Antarmuka Penumpang .....	92
<b>BAB VI</b>	
<b>PENGUJIAN DAN ANALISIS .....</b>	<b>96</b>
6.1 Pengujian .....	96
6.1.1 Pengujian Validasi .....	96
6.1.1.1 Kasus Uji Validasi .....	97
6.1.1.2 Hasil Pengujian Validasi .....	103
6.1.2 Pengujian Performa .....	107
6.1.3 Pengujian <i>Usability</i> .....	108
6.2 Analisis .....	111
6.2.1 Analisis Hasil Pengujian Validasi .....	111
6.2.2 Analisis Hasil Pengujian Performa .....	111
6.2.3 Analisis Hasil Pengujian <i>Usability</i> .....	111
<b>BAB VII</b>	
<b>PENUTUP .....</b>	<b>113</b>

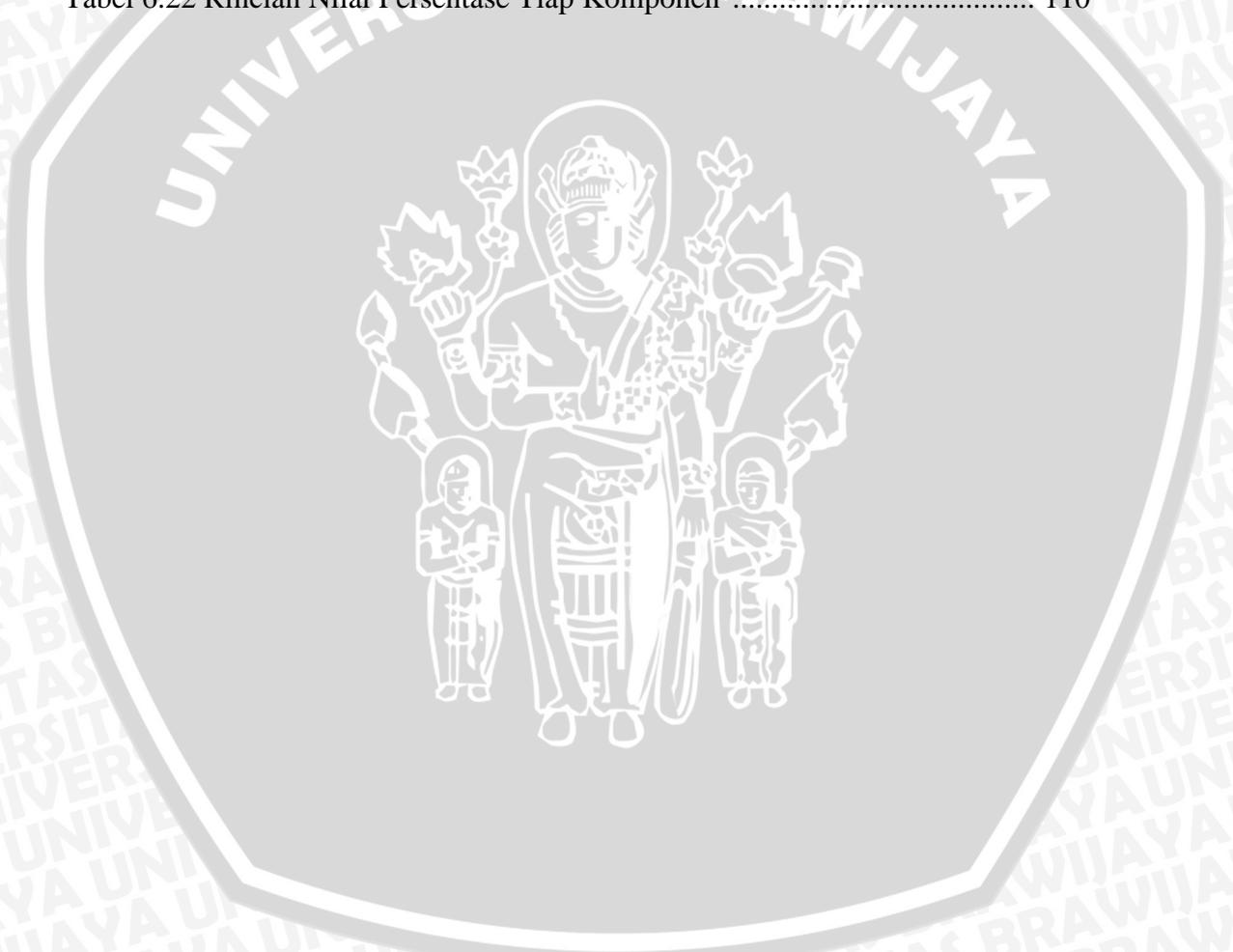
7.1 Kesimpulan .....	113
7.2 Saran .....	113
<b>DAFTAR PUSTAKA .....</b>	<b>115</b>
<b>LAMPIRAN 1 .....</b>	<b>117</b>



## DAFTAR TABEL

Tabel 2.1	Keterangan Simbol-Simbol <i>Use Case Diagram</i> .....	7
Tabel 2.2	Keterangan Simbol-Simbol <i>Sequence Diagram</i> .....	11
Tabel 2.3	Keterangan Simbol-Simbol <i>ERD Diagram</i> .....	15
Tabel 4.1	Identifikasi Aktor .....	30
Tabel 4.2	Spesifikasi Kebutuhan Penumpang .....	31
Tabel 4.3	Spesifikasi Kebutuhan Admin .....	32
Tabel 4.4	Skenario <i>Use Case</i> Melihat Info Bus Melalui SMS .....	33
Tabel 4.5	Skenario <i>Use Case</i> Melihat Info Bus Melalui <i>Website</i> .....	34
Tabel 4.6	Skenario <i>Use Case</i> Memesan Tiket Melalui SMS .....	35
Tabel 4.7	Skenario <i>Use Case</i> Memesan Tiket Melalui <i>Website</i> .....	36
Tabel 4.8	Skenario <i>Use Case</i> Mengkonfirmasi Pembayaran Melalui SMS ....	37
Tabel 4.9	Skenario <i>Use Case Login</i> .....	38
Tabel 4.10	Skenario <i>Use Case</i> Menjalankan Service SMS GATEWAY .....	39
Tabel 4.11	Skenario <i>Use Case</i> Mengelola Data Pengemudi .....	40
Tabel 4.12	Skenario <i>Use Case</i> Mengelola Data Bus .....	41
Tabel 4.13	Skenario <i>Use Case</i> Mengelola Data Jadwal Bus .....	42
Tabel 4.14	Skenario <i>Use Case</i> Mengelola Data Pemesanan .....	43
Tabel 4.15	Skenario <i>Use Case</i> Mengelola Data <i>Users</i> .....	45
Tabel 4.16	Struktur Tabel Admin .....	46
Tabel 4.17	Struktur Tabel Jadwal .....	46
Tabel 4.18	Struktur Tabel Pengemudi .....	46
Tabel 4.19	Struktur Tabel Bus .....	46
Tabel 4.20	Struktur Tabel Pemesan .....	47
Tabel 5.1	Spesifikasi Perangkat Keras Komputer .....	75
Tabel 5.2	Spesifikasi Perangkat Lunak Komputer .....	75
Tabel 5.3	Struktur Tabel Admin .....	76
Tabel 5.4	Struktur Tabel Jadwal .....	77
Tabel 5.5	Struktur Tabel Pengemudi .....	77
Tabel 5.6	Struktur Tabel Bus .....	77
Tabel 5.7	Struktur Tabel Pemesan .....	78
Tabel 6.1	Kasus Uji Untuk Pengujian Validasi <i>Login Sah</i> .....	97
Tabel 6.2	Kasus Uji Untuk Pengujian Validasi <i>Login Tidak Sah</i> .....	97
Tabel 6.3	Kasus Uji Untuk Pengujian Validasi <i>Logout</i> .....	97
Tabel 6.4	Kasus Uji Untuk Pengujian Validasi Menjalankan Service SMS GATEWAY .....	98
Tabel 6.5	Kasus Uji Untuk Pengujian Validasi Mengelola Data Pengemudi..	98
Tabel 6.6	Kasus Uji Untuk Pengujian Validasi Mengelola Data Bus .....	99
Tabel 6.7	Kasus Uji Untuk Pengujian Validasi Mengelola Data Jadwal .....	99
Tabel 6.8	Kasus Uji Untuk Pengujian Validasi Mengelola Data Pemesanan..	99
Tabel 6.9	Kasus Uji Untuk Pengujian Validasi Mengelola Data <i>Users</i> .....	100
Tabel 6.10	Kasus Uji Untuk Pengujian Validasi Melihat Info Bus Via SMS Sah .....	100
Tabel 6.11	Kasus Uji Untuk Pengujian Validasi Melihat Info Bus Via SMS Tidak Sah .....	101
Tabel 6.12	Kasus Uji Untuk Pengujian Validasi Melihat Info Bus Via <i>Website</i>	101

Tabel 6.13 Kasus Uji Untuk Pengujian Validasi Memesan Tiket Via SMS Sah .....	101
Tabel 6.14 Kasus Uji Untuk Pengujian Validasi Memesan Tiket Via SMS Tidak Sah .....	102
Tabel 6.15 Kasus Uji Untuk Pengujian Validasi Memesan Tiket Via Website .....	102
Tabel 6.16 Kasus Uji Untuk Pengujian Validasi Mengkonfirmasi Pembayaran Sah .....	102
Tabel 6.17 Kasus Uji Untuk Pengujian Validasi Mengkonfirmasi Pembayaran Tidak Sah .....	103
Tabel 6.18 Hasil Pengujian Validasi .....	103
Tabel 6.19 Hasil Pengujian Performa .....	107
Tabel 6.20 Komponen dan Pertanyaan Pengujian <i>Usability</i> .....	109
Tabel 6.21 Hasil Pengujian <i>Usability</i> .....	110
Tabel 6.22 Rincian Nilai Persentase Tiap Komponen .....	110



## DAFTAR GAMBAR

Gambar 2.1	Diagram UML .....	6
Gambar 2.2	Contoh DFD .....	9
Gambar 2.3	Relasi Basis Data .....	12
Gambar 3.1	Diagram Blok Sistem .....	24
Gambar 4.1	Diagram Blok Perancangan .....	26
Gambar 4.2	Sistem Pemesanan Saat Ini .....	27
Gambar 4.3	Sistem Pemesanan Berbasis SMS GATEWAY .....	28
Gambar 4.4	Diagram <i>Use Case</i> Penumpang .....	33
Gambar 4.5	Diagram Aktifitas Melihat Info Bus Melalui SMS .....	34
Gambar 4.6	Diagram Aktifitas Melihat Info Bus Melalui <i>Website</i> .....	35
Gambar 4.7	Diagram Aktifitas Memesan Tiket Melalui SMS .....	36
Gambar 4.8	Diagram Aktifitas Memesan Tiket Melalui <i>Website</i> .....	36
Gambar 4.9	Diagram Aktifitas Mengkonfirmasi Pembayaran Melalui SMS..	37
Gambar 4.10	Diagram <i>Use Case</i> Admin .....	38
Gambar 4.11	Diagram Aktifitas <i>Login</i> .....	39
Gambar 4.12	Diagram Aktifitas Menjalankan Service SMS GATEWAY.....	40
Gambar 4.13	Diagram Aktifitas Mengelola Data Pengemudi .....	41
Gambar 4.14	Diagram Aktifitas Mengelola Data Bus .....	42
Gambar 4.15	Diagram Aktifitas Mengelola Data Jadwal Bus .....	43
Gambar 4.16	Diagram Aktifitas Mengelola Data Pemesanan .....	44
Gambar 4.17	Diagram Aktifitas Mengelola Data Admin .....	45
Gambar 4.18	Diagram <i>Entity Relationship</i> Basis Data .....	45
Gambar 4.19	DFD Level 0 Proses Reservasi Tiket .....	48
Gambar 4.20	DFD Level 1 Proses Reservasi Tiket .....	48
Gambar 4.21	DFD Level 1 (lanjutan) Proses Mengelola Sistem Reservasi ....	49
Gambar 4.22	DFD Level 1 (lanjutan) Proses Mengelola Data Setting SMS ....	50
Gambar 4.23	DFD Level 1 (lanjutan) Proses Mengelola Data Bus .....	51
Gambar 4.24	DFD Level 1 (lanjutan) Proses Mengelola Data Pengemudi .....	52
Gambar 4.25	DFD Level 1 (lanjutan) Proses Mengelola Data Jadwal .....	52
Gambar 4.26	DFD Level 1 (lanjutan) Proses Mengelola Data Pemesanan.....	53
Gambar 4.27	DFD Level 1 (lanjutan) Proses Mengelola Data <i>Users</i> .....	53
Gambar 4.28	DFD Level 1 (lanjutan) Proses Melakukan Reservasi .....	53
Gambar 4.29	Perancangan Alur Kerja Melihat Info Via SMS .....	54
Gambar 4.30	Perancangan Alur Kerja Melihat Info Via <i>Website</i> .....	55
Gambar 4.31	Perancangan Alur Kerja Memesan Tiket Via SMS .....	55
Gambar 4.32	Perancangan Alur Kerja Memesan Tiket Via <i>Website</i> .....	56
Gambar 4.33	Perancangan Alur Kerja Mengkonfirmasi Pembayaran Via SMS .....	56
Gambar 4.34	Tampilan Antarmuka SMS Melihat Info .....	58
Gambar 4.35	Tampilan Antarmuka SMS Informasi Bus .....	58
Gambar 4.36	Tampilan Antarmuka SMS Memesan Tiket .....	58
Gambar 4.37	Tampilan Antarmuka SMS Konfirmasi Pemesanan Tiket Berhasil .....	59
Gambar 4.38	Tampilan Antarmuka SMS Konfirmasi Bayar .....	60
Gambar 4.39	Perancangan Alur Kerja <i>Login</i> .....	60

Gambar 4.40 Perancangan Alur Kerja Menjalankan Service SMS GATEWAY .....	61
Gambar 4.41 Perancangan Alur Kerja Lihat Pengemudi .....	61
Gambar 4.42 Perancangan Alur Kerja Tambah Pengemudi .....	62
Gambar 4.43 Perancangan Alur Kerja Ubah Pengemudi .....	62
Gambar 4.44 Perancangan Alur Kerja Hapus Pengemudi .....	63
Gambar 4.45 Perancangan Alur Kerja Lihat Bus .....	63
Gambar 4.46 Perancangan Alur Kerja Tambah Bus .....	63
Gambar 4.47 Perancangan Alur Kerja Ubah Bus .....	64
Gambar 4.48 Perancangan Alur Kerja Hapus Bus .....	64
Gambar 4.49 Perancangan Alur Kerja Lihat Jadwal Bus .....	65
Gambar 4.50 Perancangan Alur Kerja Tambah Jadwal Bus .....	65
Gambar 4.51 Perancangan Alur Kerja Lihat Pemesanan .....	66
Gambar 4.52 Perancangan Alur Kerja Konfirmasi Pemesanan .....	67
Gambar 4.53 Perancangan Alur Kerja Lihat <i>Users</i> .....	67
Gambar 4.54 Perancangan Alur kerja Tambah <i>Users</i> .....	68
Gambar 4.55 Tampilan Antarmuka Halaman <i>Login</i> .....	68
Gambar 4.56 Tampilan Antarmuka Halaman <i>Home</i> .....	69
Gambar 4.57 Tampilan Antarmuka Halaman <i>Run SMS</i> .....	70
Gambar 4.58 Tampilan Antarmuka Halaman Pengemudi .....	70
Gambar 4.59 Tampilan Antarmuka Halaman Bus .....	71
Gambar 4.60 Tampilan Antarmuka Halaman Jadwal .....	71
Gambar 4.61 Tampilan Antarmuka Halaman Pemesanan .....	72
Gambar 4.62 Tampilan Antarmuka Halaman <i>Users</i> .....	73
Gambar 5.1 Pohon Implementasi .....	74
Gambar 5.2 Diagram Konseptual .....	76
Gambar 5.3 Implementasi Proses <i>Login</i> .....	78
Gambar 5.4 Implementasi Proses Mengelola Data Pengemudi .....	80
Gambar 5.5 Implementasi Alur Kerja Admin Mengelola Data Bus .....	82
Gambar 5.6 Implementasi Alur Kerja Admin Mengelola Jadwal Bus .....	85
Gambar 5.7 Implementasi Alur Kerja Admin Mengelola Data Pemesanan ..	87
Gambar 5.8 Tampilan Antarmuka Halaman <i>Login</i> .....	89
Gambar 5.9 Tampilan Antarmuka Halaman Utama .....	89
Gambar 5.10 Tampilan Antarmuka Halaman <i>Run SMS</i> .....	89
Gambar 5.11 Tampilan Antarmuka Halaman Pengemudi .....	90
Gambar 5.12 Tampilan Antarmuka Halaman Bus .....	90
Gambar 5.13 Tampilan Antarmuka Halaman Jadwal .....	91
Gambar 5.14 Tampilan Antarmuka Halaman Pemesanan .....	91
Gambar 5.15 Tampilan Antarmuka Halaman <i>Users</i> .....	92
Gambar 5.16 Tampilan Antarmuka SMS Melihat Info .....	92
Gambar 5.17 Tampilan Antarmuka SMS Informasi Bus .....	93
Gambar 5.18 Tampilan Antarmuka SMS Memesan Tiket .....	93
Gambar 5.19 Tampilan Antarmuka SMS Konfirmasi Pemesanan Tiket .....	94
Gambar 5.20 Tampilan Antarmuka SMS Konfirmasi Pemesanan Tiket Diterima .....	94
Gambar 5.21 Tampilan Antarmuka SMS Konfirmasi Pembayaran .....	95
Gambar 6.1 Pohon Pengujian dan Analisis .....	96

## DAFTAR LAMPIRAN

Lampiran 1 : Kuisioner Pengujian Usability Pada Prototipe Sistem Reservasi  
Tiket Bus Menggunakan PHP dan SMS GATEWAY  
(Studi Kasus di Terminal Arjosari Malang). ..... 117



# BAB I

## PENDAHULUAN

### 1.1 LATAR BELAKANG

Perjalanan dengan menggunakan alat transportasi darat yaitu bus sangat digemari oleh penduduk Indonesia karena harganya yang terjangkau. Hal ini mendorong para pecinta bus untuk semakin mengembangkan sebuah aplikasi agar dapat membantu kerja manusia atau bahkan melebihi kemampuan kerja manusia.

Salah satu penerapannya adalah dalam bidang pemesanan tiket perjalanan menggunakan bus. Umumnya pemesanan tiket bus dilakukan secara manual yaitu langsung mendatangi tempat yang menjual tiket perjalanan bus. Tetapi dengan menggunakan aplikasi ini memudahkan pembeli dalam melakukan pemesanan tiket perjalanan bus. Aplikasi yang dibuat ini adalah aplikasi pemesanan tiket bus untuk membantu dalam melakukan pemesanan tiket bus dan pembeli akan mendapatkan *report* melalui *SMS Gateway*. Aplikasi ini dibuat dengan melakukan studi lapangan di Terminal Arjosari Malang. Pada aplikasi ini menggunakan *SMS Gateway* untuk mengirimkan *report* kepada pembeli setelah melakukan pemesanan tiket bus secara *online*.

*SMS Gateway* pada aplikasi ini berfungsi sebagai penghubung yang melakukan *delay sms* antara *External Short Message Entity* (ESME) dan *Short Message Service Center* (SMSC) dan sebaliknya. Komunikasi antara ESME dan *SMS Gateway* dapat menggunakan protokol *Short Message Peer to Peer* (SMPP) atau dengan HTTP, sementara ke SMSC menggunakan SMPP [DAV-09].

Proses sistem reservasi tiket bus ini sebagai berikut : calon pembeli yang ingin melakukan transaksi dapat membuka *website* dan melihat info serta layanan yang tersedia. Jika calon pembeli sudah melihat info keberangkatan bus, proses selanjutnya calon pembeli dapat melakukan pemesanan melalui SMS atau mengisi *form* pembelian tiket yang telah disediakan. Pemesanan melalui SMS dilakukan dengan cara mengirimkan SMS berisikan data data pemesanan tiket bus (nama bus, tujuan, tanggal keberangkatan dan jumlah tiket). Jika data telah diterima maka aplikasi akan mengirimkan pesan kepada pembeli tentang status pemesanan tiket, misal nomer tempat duduk yang diinginkan telah dipesan orang lain, dll.

Untuk pemesanan tiket dengan mengisi *form* dapat dilakukan langsung pada saat calon pembeli membuka *website* reservasi tiket bus.

Jika data yang telah dimasukkan pembeli masuk pada basis data maka pembeli akan mendapatkan pesan bahwa tiket yang dipesan tersedia atau tidak tersedia. Proses pembayaran untuk pemesanan tiket via SMS atau mengisi *form* pembelian tiket *online* pada sistem ini dilakukan dengan mengirimkan sejumlah uang sesuai dengan total pembelian ke nomer rekening yang telah disediakan dan calon pembeli harus menyimpan bukti transfer pembayaran sebagai bukti untuk menukar dengan tiket yang sudah dipesan sebelumnya. Dan skripsi ini lebih ditekankan pada penggunaan *SMS Gateway* dalam aplikasi pemesanan tiket bus secara *online*.

## 1.2 RUMUSAN MASALAH

Berdasarkan permasalahan yang telah dijelaskan pada latar belakang, maka rumusan masalah dapat disusun sebagai berikut :

1. Merancang sistem reservasi tiket *online* yang dikendalikan oleh *SMS Gateway* yang terdiri dari analisis kebutuhan, perancangan data dan perancangan antarmuka dari sistem.
2. Mengimplementasikan sistem reservasi tiket *online* yang dikendalikan oleh *SMS Gateway*.
3. Menguji sistem reservasi tiket *online* yang dikendalikan oleh *SMS Gateway* meliputi pengujian validasi, pengujian performa dan pengujian *usability*.

## 1.3 BATASAN MASALAH

Agar permasalahan yang dirumuskan lebih terfokus dan tidak terjadi pelebaran topik, maka penelitian proposal judul ini dibatasi dalam hal:

1. Aplikasi ini menyediakan layanan pemesanan perjalanan tiket bus melalui *SMS Gateway* dan info perjalanan bus yang tersedia di Terminal Arjosari Malang.

2. Pembuatan aplikasi ini menggunakan SMS Gateway GAMMU dengan menggunakan modem GSM Huawei E 160.
3. Pembuatan aplikasi menggunakan bahasa pemrograman PHP dan implementasi basis data MySQL.
4. Pembuatan aplikasi ini dilatarbelakangi oleh calon penumpang yang sudah sering melakukan pemesanan tiket bus di terminal secara manual.

#### 1.4 TUJUAN PENELITIAN

Tujuan akhir dari penelitian ini adalah merancang dan mengembangkan sistem reservasi tiket *online* dengan menggunakan *SMS Gateway*. Aplikasi yang dibangun digunakan untuk memudahkan calon penumpang angkutan bus di Terminal Arjosari mendapatkan tiket dan layanan-layanan lainnya yang tersedia.

#### 1.5 MANFAAT PENELITIAN

Manfaat yang diperoleh melalui pengerjaan skripsi ini adalah :

- a. Bagi penulis
  - Menerapkan ilmu yang telah didapatkan di Teknik Informatika konsentrasi Rekayasa Perangkat Lunak Universitas Brawijaya.
  - Mendapatkan pemahaman lebih lanjut dalam pengembangan Reservasi Tiket Bus Menggunakan PHP dan *SMS Gateway* (Studi Kasus Terminal Arjosari Malang).
- b. Bagi Pengguna
  - Memberikan kemudahan kepada calon penumpang angkutan bus di terminal Arjosari Malang untuk mendapatkan tiket dan layanan-layanan lainnya yang tersedia.

#### 1.6 SISTEMATIKA PENULISAN

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 dan sistematika penulisan.

## **BAB II Dasar Teori**

Membahas teori-teori yang mendukung dalam pengembangan dan perancangan aplikasi.

## **BAB III Metodologi**

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

## **BAB IV Analisis Kebutuhan dan Perancangan**

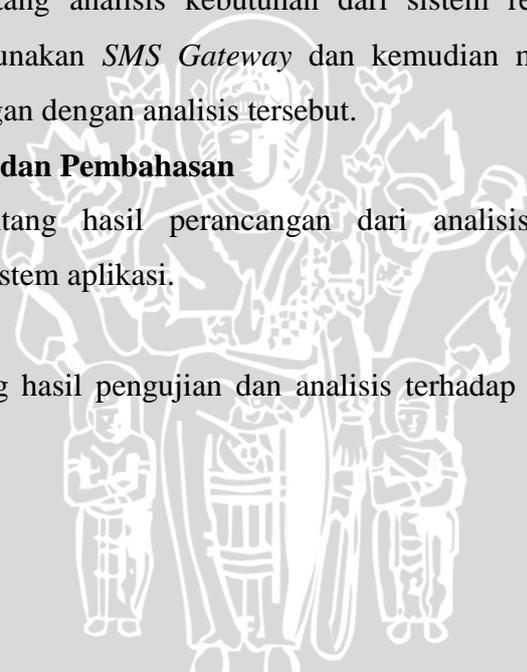
Membahas tentang analisis kebutuhan dari sistem reservasi tiket bus dengan menggunakan *SMS Gateway* dan kemudian merancang hal-hal yang berhubungan dengan analisis tersebut.

## **BAB V Implementasi dan Pembahasan**

Membahas tentang hasil perancangan dari analisis kebutuhan dan implementasi sistem aplikasi.

## **BAB VI Pengujian**

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



## BAB II

### DASAR TEORI

#### 2.1 REKAYASA PERANGKAT LUNAK

Rekayasa perangkat lunak (*software engineering*) merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin [ROS-11 :4]. Rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria berikut [ROS-11 :4]:

- Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (*maintainable*)
- Dapat diandalkan dengan proses bisnis yang dijalankan dan perubahan yang terjadi (*dependable* dan *robust*)
- Efisien dari segi sumber daya dan penggunaan
- Kemampuan untuk dipakai sesuai dengan kebutuhan

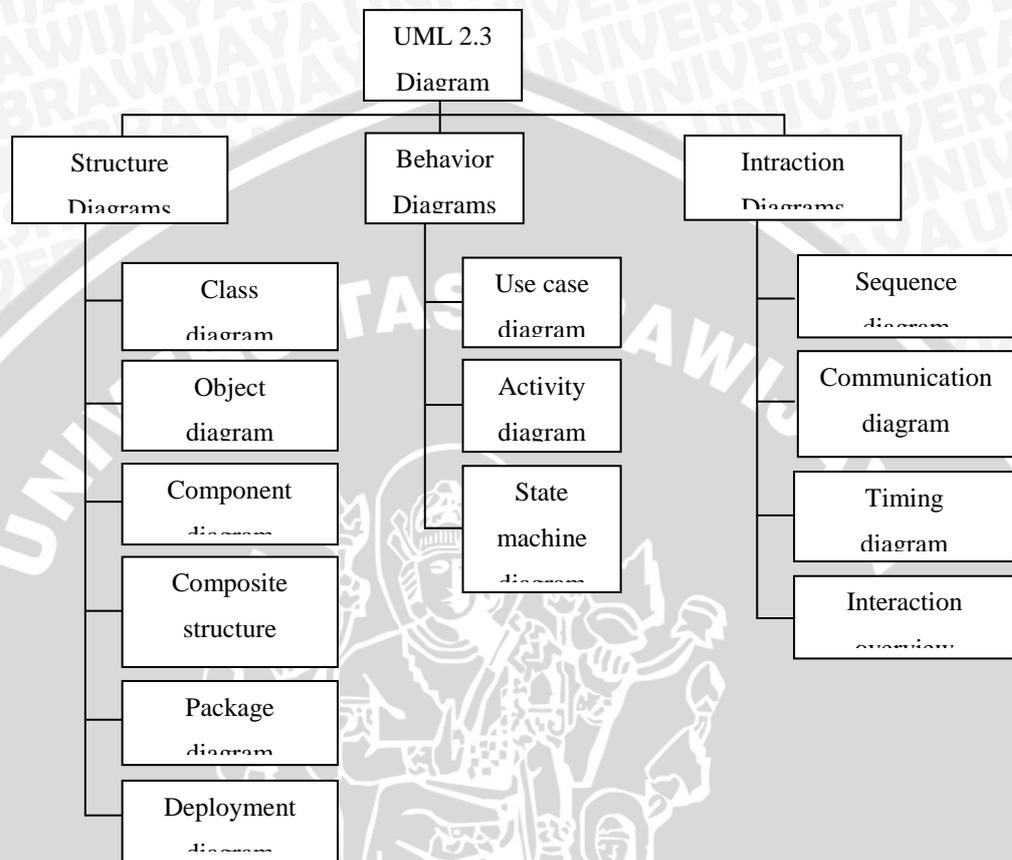
Dari kriteria diatas maka perangkat lunak yang baik adalah perangkat lunak yang dapat memenuhi kebutuhan pelanggan (*customer*) atau pemakai (*user*) perangkat lunak atau berorientasi pada pelanggan atau pemakai perangkat lunak, bukan berorientasi pada pembuat atau pengembang perangkat lunak.

Pada skripsi ini digunakan metode analisis kebutuhan, perancangan, implementasi, dan pengujian berorientasi objek menggunakan bahasa pemodelan *Unified Modelling Language* (UML). UML adalah bahasa untuk menggambarkan (*visualizing*), menspesifikasikan (*specifying*), membangun (*constructing*), dan mendokumentasikan (*documenting*) artefak dari sebuah sistem perangkat lunak [BOO-05].

#### 2.2 UNIFIED MODELLING LANGUAGE

*Unified Modelling Language* (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram teks-teks pendukung. UML muncul karena adanya kebutuhan pemodelan

visual untuk menspesifikan, menggambarkan, membangun, dan mendokumentasikan dari sistem perangkat lunak [ROS-11:117].



Gambar 2.1 Diagram UML

Sumber : [ROS-11:121]

*Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan system atau rangkaian perubahan yang terjadi pada sebuah sistem. *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi system dengan system lain maupun interaksi antar subsistem pada suatu sistem [ROS-11 :121].

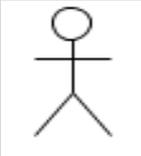
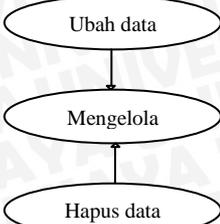
### 2.2.1 USE CASE DIAGRAM

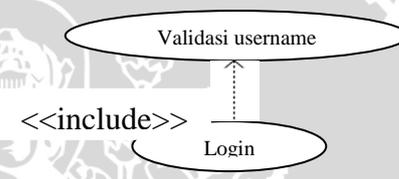
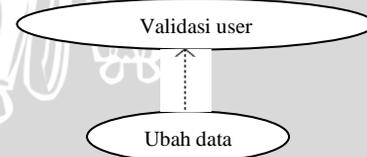
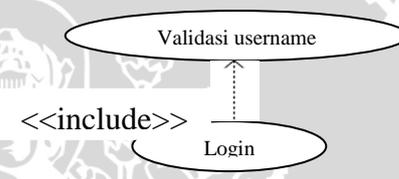
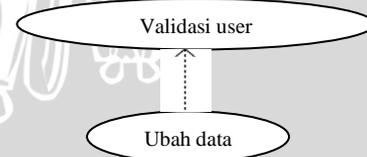
Diagram *Use case* merupakan pemodelan untuk kelakuan (*behavior*) system informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Diagram *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah system informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu [ROS-11:130].

Syarat penamaan pada diagram *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

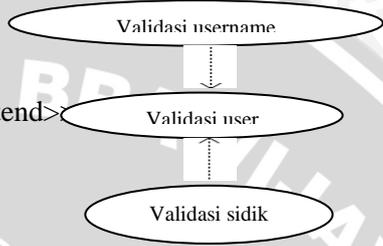
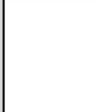
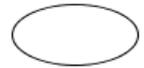
1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Simbol aktor adalah gambar orang, namun aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antarunit atau aktor.

Tabel 2.1 Keterangan simbol - simbol *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		Aktor/ <i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan system informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
2		Generalisasi/ <i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya : 

			<p>Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum).</p>
<p>3</p>	 <p>Menggunakan / <i>include / uses</i></p> <p><code>&lt;&lt;include&gt;&gt;</code></p> <p><code>&lt;&lt;uses&gt;&gt;</code></p>	<p>Fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>Ada dua sudut pandang yang cukup besar mengenai include di <i>use case</i> :</p> <p>Include berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, misal pada kasus berikut :</p>  <p>Include berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, missal pada kasus berikut :</p>  <p>Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>	<p>Fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>Ada dua sudut pandang yang cukup besar mengenai include di <i>use case</i> :</p> <p>Include berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, misal pada kasus berikut :</p>  <p>Include berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, missal pada kasus berikut :</p>  <p>Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>



4		<p>Ekstensi/ <i>Extend</i>                  &lt;&lt;extend&gt;&gt;</p>	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misal</p> <div style="text-align: center;">  </div> <p>&lt;&lt;extend&gt;&gt;</p> <p>Arah panah mengarah pada <i>use case</i> yang ditambahkan.</p>
5		<p>Asosiasi/ <i>Association</i></p>	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interksi dengan aktor.</p>
6		<p><i>System</i></p>	<p>Menspesifikasikan paket yang menampilkan sistem secara terbatas.</p>
7		<p><i>Use Case</i></p>	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>.</p>
8		<p><i>Collaboration</i></p>	<p>Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).</p>

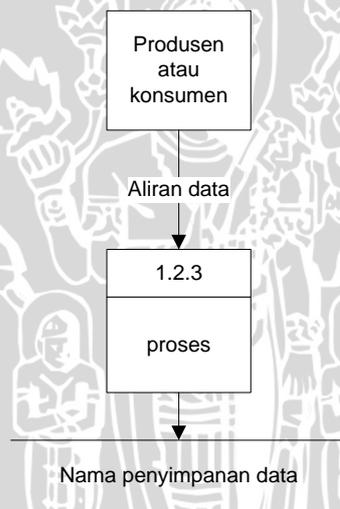


9		Note	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi
---	-----------------------------------------------------------------------------------	------	-----------------------------------------------------------------------------------------------

Sumber : [ROS-11 :130]

### 2.2.2 Data flow diagram

*Data Flow Diagram* (DFD) awalnya dikembangkan oleh Chris Gane dan Trish Sarson pada tahun 1979 yang termasuk dalam *Structured Systems Analysis and Design Methodology* (SSADM) yang ditulis oleh Chris Gane dan Trish Sarson. Sistem yang dikembangkan ini berbasis pada dekomposisi fungsionalitas dari sebuah sistem. Berikut adalah contoh DFD yang dikembangkan oleh Chris Gane dan Trish Sarson :



**Gambar 2.2 Contoh DFD**

Sumber : [ROS-11:63]

Edward Yourdon dan Tom DeMarco memperkenalkan metode yang lain pada tahun 1980-an di mana mengubah persegi dengan sudut lengkung (pada DFD Chris Gane dan Trish Sarson) dengan lingkaran untuk menotasikan. DFD Edward Yourdon dan Tom DeMarco populer digunakan sebagai model analisis sistem perangkat lunak untuk sistem perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.

Informasi yang ada di dalam perangkat lunak dimodifikasi dengan beberapa transformasi yang dibutuhkan. DFD atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD dapat dibagi menjadi beberapa level yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur.

DFD tidak sesuai untuk memodelkan sistem perangkat lunak yang akan dibangun menggunakan pemrograman berorientasi objek. Paradigma pemrograman terstruktur dan pemrograman berorientasi objek merupakan hal yang berbeda. Jangan mencampur adukkan pemrograman terstruktur dan pemrograman berorientasi objek [ROS-11:64].

### 2.2.3 Sequence Diagram

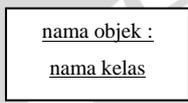
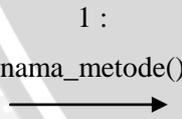
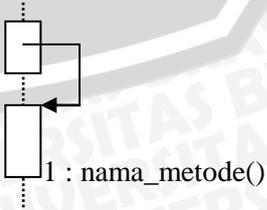
*Sequence diagram* atau diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Penggambaran diagram sekuen harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu [ROS-11:137].

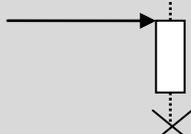
Diagram sekuen yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri. Banyaknya *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

Berikut adalah simbol-simbol yang ada pada diagram sekuen :

Tabel 2.2 Keterangan simbol - simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
----	--------	------	------------

1		<p>Aktor/ <i>Actor</i></p> <p>atau</p> <p>tanpa waktu aktif</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan system informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor</p>
2		<p>Garis hidup/ <i>Lifeline</i></p>	<p>Menyatakan kehidupan suatu objek</p>
3		<p>Objek/ <i>Object</i></p>	<p>Menyatakan objek yang berinteraksi pesan</p>
4		<p>Waktu aktif</p>	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.</p>
5		<p>Pesan tipe <i>create</i></p>	<p>Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.</p>
6		<p>Pesan tipe <i>call</i></p>	<p>Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lin atau dirinya sendiri</p> 

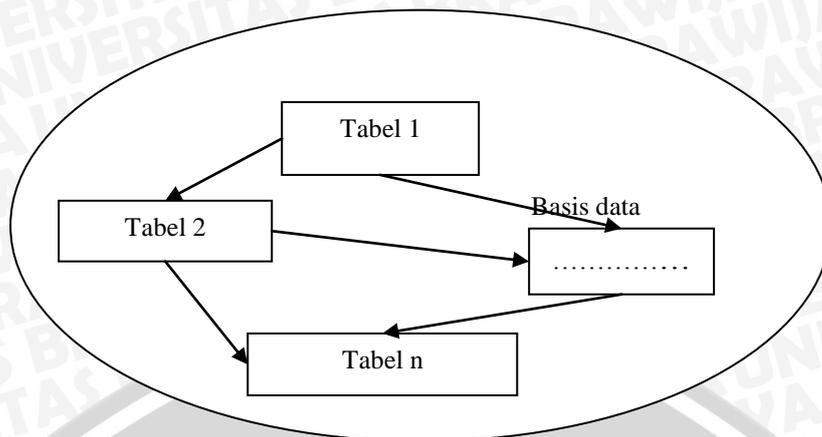
			Arah panah mengarah pada objek yang memiliki operasi / metode, karena ini memanggil operasi / metode maka operasi / metode yang dipanggil harus ada pada diagram kelas sesuai dengan objek yang berinteraksi.
7	<p>1 : masukan</p> 	Pesan tipe <i>send</i>	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang akan dikirim
8	<p>1 : keluaran</p> 	Pesan tipe <i>return</i>	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
9	<p>&lt;&lt;destroy&gt;&gt;</p> 	Pesan tipe <i>destroy</i>	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy.

Sumber : [ROS-11:138]

Penomoran pesan berdasarkan urutan interaksi pesan. Penggambaran letak pesan harus berurutan, pesan yang lebih atas dari lainnya adalah pesan yang berjalan terlebih dahulu [ROS-11 :137].

### 2.3 BASIS DATA

Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat. Pada bab ini menggunakan basis data relasional yang diimplementasikan dengan tabel-tabel yang saling memiliki relasi seperti pada gambar berikut.



**Gambar 2.3 Relasi Basis Data**

Sumber : [ROS-11 :44]

Sistem informasi tidak dapat dipisahkan dengan kebutuhan akan basis data apa pun bentuknya, entah berupa *file* teks maupun *Database Management System* (DBMS) [ROS-11 :44].

Kebutuhan basis data dalam sistem informasi meliputi :

- Memasukkan, menyimpan, dan mengambil data
- Membuat laporan berdasarkan data yang telah disimpan

### 2.3.1 DBMS

*Database Management System* (DBMS) atau dalam bahasa Indonesia sering disebut sebagai Sistem Manajemen Basis Data adalah suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola, dan menampilkan data. Suatu sistem aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut :

- Menyediakan fasilitas untuk mengelola akses data
- Mampu menangani integritas data
- Mampu menangani akses data yang dilakukan
- Mampu menangani *backup* data

Karena pentingnya data bagi suatu organisasi/perusahaan, maka hampir sebagian besar perusahaan memanfaatkan DBMS dalam mengelola data yang mereka miliki. Pengelolaan DBMS sendiri biasanya ditangani oleh tenaga ahli yang spesialis menangani DBMS yang disebut sebagai *Database Administrator* (DBA) [ROS-11 :45].

## 2.4 SQL

*Structured Query Language* (SQL) adalah bahasa yang digunakan untuk mengelola data pada *Relational* DBMS atau RDBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus [ROS-11 :46].

SQL mulai berkembang pada tahun 1970an. SQL mulai digunakan sebagai standar yang resmi pada tahun 1986 oleh *American National Standards Institute* (ANSI) dan pada tahun 1987 oleh *International Organization for Standardization* (ISO) dan disebut sebagai SQL-86. Pada perkembangannya, SQL beberapa kali dilakukan revisi.

Meskipun SQL diadopsi dan diakui sebagai bahasa standar oleh hampir sebagian besar RDBMS yang beredar saat ini, tetapi tidak semua standar yang tercantum dalam SQL diimplementasikan oleh seluruh DBMS tersebut. Sehingga kadang-kadang ada perbedaan perilaku (hasil yang ditampilkan) oleh DBMS yang berbeda padahal *query* yang dimasukkan sama.

Berikut ini adalah contoh pengaksesan data pada DBMS dengan SQL yang secara umum terdiri dari 4 hal sebagai berikut :

### 1. Memasukkan data (*insert*)

```
INSERT INTO Tabel_mahasiswa
      (nim, nama, tanggal_lahir)
VALUES
      ('1351058', 'Rosa', '1986-01-01');
```

*Query* di atas digunakan untuk memasukkan data mahasiswa dengan NIM 13501058, nama Rosa, dan tanggal lahir 1 Januari 1986 ke tabel "Tabel\_mahasiswa".

### 2. Mengubah data (*update*)

```
Update Tabel_mahasiswa
      SET
      Tanggal_lahir = '1990-03-04'
      WHERE
      nim = '13501058';
```

*Query* di atas digunakan untuk mengubah data tanggal lahir mahasiswa dengan NIM = 13501058 menjadi 4 Maret 1990 dalam tabel “Tabel\_mahasiswa”.

**3. Menghapus data (*delete*)**

**DELETE FROM** Tabel\_mahasiswa

**WHERE**

nim = ‘13501058’;

*Query* di atas digunakan untuk menghapus data mahasiswa dengan NIM = 13501058 dari tabel “Tabel\_mahasiswa”.

**4. Menampilkan data (*select*)**

**SELECT** nim,

Nama

**FROM** Tabel\_mahasiswa

**WHERE**

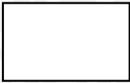
Nim = ‘13501058’;

*Query* di atas digunakan untuk menampilkan data mahasiswa yang tersimpan dalam “Tabel\_mahasiswa” dengan NIM = 13501058 [ROS-11 :47].

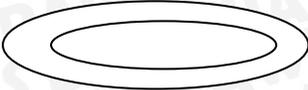
**2.5 ERD**

Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data rasional. Sehingga jika penyimpanan basis data menggunakan *Object Oriented Database Management System* (OODBMS) maka perancangan basis data tidak perlu menggunakan ERD. Berikut adalah symbol-simbol yang digunakan pada ERD :

**Tabel 2.3 Keterangan simbol-simbol ERD diagram**

Simbol	Deskripsi
Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data
Atribut 	<i>Field</i> atau kolom dat yang butuh disimpan dalam suatu entitas



<p>Atribut kunci primer</p> 	<p>Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id</p>
<p>Atribut multinilai / <i>multivalue</i></p> 	<p>Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu</p>
<p>Relasi</p> 	<p>Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja</p>
<p>Asosiasi / association</p> 	<p>Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian</p>

Sumber : [ROS-11 :49]

## 2.6 PHP

PHP merupakan script untuk pemrograman *script web server-side*, *script* yang membuat dokumen HTML secara *on the fly*, dokumen yang HTML yang dihasilkan dari suatu aplikasi yang bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML [SID-04].

Dengan menggunakan PHP maka *maintenance* suatu situs *web* menjadi lebih mudah. Proses *update* data dapat dilakukan dengan menggunakan aplikasi yang dibuat dengan menggunakan *script* PHP [PAN-09].

PHP/FI merupakan nama awal dari PHP. PHP-*Personal Home Page*, FI adalah *Form Interface*. Dibuat pertama kali oleh *Rasmus Lerdoff*. PHP, awalnya merupakan program CGI yang dikhususkan untuk menerima *input* melalui *form* yang ditampilkan dalam *browser web*. *Software* ini disebarkan dan dilisensikan sebagai perangkat lunak *Open Source*. *PHP* secara resmi merupakan kependekan dari PHP:*Hypertext Preprocessor*, merupakan *script server side* yang disisipkan pada HTML [SID-04].

## 2.7 SMS GATEWAY

repository.ub.ac.id

*Short Message Service* (SMS) adalah protokol layanan pertukaran pesan text singkat (sebanyak 160 karakter per pesan) antar telepon. SMS ini pada awalnya adalah bagian dari standar teknologi seluler GSM, yang kemudian juga tersedia di teknologi CDMA, telepon rumah PSTN, dan lainnya.

SMS *Gateway* merupakan pintu gerbang bagi penyebaran Informasi dengan menggunakan SMS. *SMS Gateway* dapat menyebarkan pesan ke ratusan nomor secara otomatis dan cepat yang langsung terhubung dengan database nomor ponsel saja tanpa harus mengetik ratusan nomor dan pesan di ponsel anda karena semua nomor akan diambil secara otomatis dari database tersebut. Layanan SMS sangat populer dan sering dipakai oleh pengguna *handphone*. SMS menyediakan pengiriman pesan text secara cepat, mudah dan murah. Kini SMS tidak terbatas untuk komunikasi antar manusia pengguna saja, namun juga bisa dibuat otomatis dikirim/diterima oleh peralatan (komputer, mikrokontroler, dsb) untuk mencapai suatu tujuan tertentu [MAS-11].

## 2.8 GAMMU

Gammu adalah sebuah aplikasi cross-platform yang digunakan untuk menjembatani / mengomunikasikan antara database SMS Gateway dengan sms devices. Aplikasi Gammu berupa daemon yang berjalan secara background. Setiap saat, gammu memonitor sms devices dan database sms gateway. Saat ada sms masuk ke sms devices, maka gammu langsung memindahkannya ke dalam inbox dalam database sms gateway. Sebaliknya saat Aplikasi Pengirim SMS memasukkan sms ke dalam outbox dalam database sms gateway, maka gammu mengirimkannya melalui sms devices, dan memindahkan sms ke sentitem dalam database [RAM-12].

### 2.8.1 INSTALASI DAN KONFIGURASI GAMMU

Gammu adalah aplikasi open source multi-platform, artinya gammu dapat dijalankan di sistem operasi yang bermacam-macam. Hal pertama yang harus dilakukan adalah mendapatkan paket gammu dari website <http://wammu.eu/download/gammu/> dan lakukan instalasi seperti pada

umumnya. Setelah proses instalasi selesai, pasang modem di port USB [RAM-12].

Hal terpenting dari gammu adalah file konfigurasi, contoh lengkap file konfigurasi disertakan dalam manual dan website gammu. Ada 2 file konfigurasi untuk 2 sub-aplikasi gammu. Aplikasi tersebut adalah :

1. \$ gammu
2. \$ gammu-smsd

Perintah no.1 digunakan untuk melakukan baca tulis modem secara manual, sedangkan perintah no.2 adalah aplikasi background untuk memonitor database. Perintah no.1 tidak banyak digunakan, dalam ruang lingkup penulis menggunakannya sekedar untuk mendeteksi modem.

File konfigurasi untuk aplikasi no.1 bisa bernama apa saja, untuk memudahkan, diberi nama ./gammurc

Isi dari file ./gammurc adalah :

**[gammu]**

**port = /dev/ttyUSB0**

**connection = at115200**

Port : letak modem, connection : menentukan jenis koneksi yang diinginkan. Untuk penjelasan rinci dapat dilihat di manual maupun website gammu.

Deteksi modem dan integrasinya dengan gammu menggunakan perintah : \$ sudo gammu -c ./gammurc identify dan akan dihasilkan :

**Manufacturer : Wavecom**

**Model : unknown (MULTIBAND 900E 1800)**

**Firmware : 641b09gg.Q2403A 1320676 061804 14:38**

**IMEI : 012345678901234**

**SIM IMSI : 510015300171610**

Jika sudah muncul seperti di atas, maka modem anda sudah terintegrasi dengan gammu. Untuk menjalankan gammu-smsd yang merupakan aplikasi monitor database, maka kita siapkan dulu database untuk gammu. Persiapkan user dan database dengan nama tertentu, sesuaikan struktur tabel yang ada menurut

**<http://wammu.eu/docs/manual/smsd/mysql.html>**

Setelah MySQL tersesuaikan dengan struktur gammu, atur file konfigurasi untuk aplikasi background (aplikasi no.2). File konfigurasi kita namakan ./smsdrc dan berisi :

```
[gammu]
port = /dev/ttyUSB0
connection = at115200

[smsd]
service = MYSQL #database
logfile = /var/log/smsdlog #letak log file
debuglevel = 1 #level debugging, semakin tinggi semakin dalam
commtimeout = 15
sendtimeout = 15
checksecurity = 0
user = user #user database
password = password #password database
pc = localhost #alamat database
database = sms #nama database
```

Untuk section [gammu] kita sesuaikan seperti yang ada di dalam file ./smsdrc. Section [smsd] adalah pengaturan bagaimana aplikasi background berjalan. Penjelasan lebih lanjut dapat dilihat di manual. Setelah database dan file konfigurasi tersesuaikan, maka jalankan aplikasi dengan menggunakan perintah :

```
$ gammu-smsd -c ./smsdrc
```

Untuk mengirim atau membaca SMS yang masuk, maka operasi database yang berperan. Sebagai contoh untuk mengirim SMS, maka dilakukan query :

```
INSERT INTO `sms`.`outbox` (`UpdatedInDB`, `InsertIntoDB`,
`SendingDateTime`, `Text`, `DestinationNumber`, `Coding`, `UDH`, `Class`,
`TextDecoded`, `ID`, `MultiPart`, `RelativeValidity`, `SenderID`,
`SendingTimeOut`, `DeliveryReport`, `CreatorID`) VALUES
(CURRENT_TIMESTAMP, '0000-00-00 00:00:00', '0000-00-00 00:00:00',
NULL, '08199999999999', 'Default_No_Compression', NULL, '-1', 'coba',
NULL, 'false', '-1', NULL, '0000-00-00 00:00:00', 'default', '');
```

Dimana `DestinationNumber` adalah nomor yang dituju dan `TextDecoded` adalah isi SMS

Untuk membaca SMS yang masuk, maka dilakukan query :

```
SELECT * FROM `inbox`
```

Untuk integrasi lebih lanjut, semisal pengiriman SMS secara masal, ataupun menyeleksi SMS masuk, membuat SMS autoreply, bahkan mengintegrasikan dengan aplikasi yang sudah ada semua tertuju pada database gammu.

## 2.9 PENGUJIAN PERANGKAT LUNAK

Pengujian (*Testing*) Arsitektur dari perangkat lunak berorientasi objek menghasilkan sekumpulan *layered subsystems* yang mengenkapsulasi kelas-kelas yang berkolaborasi. Setiap elemen sistem (subsistem dan *class*) melakukan fungsi yang membantu untuk mencapai kebutuhan sistem. Setiap elemen sistem sangat penting untuk menguji sebuah *object oriented sistem* pada berbagai macam level yang berbeda dalam sebuah usaha untuk menemukan kesalahan-kesalahan yang mungkin terjadi dari kolaborasi kelas-kelas dan komunikasi subsistem melewati *architetural layer* [PRE-01:631].

### 2.9.1 Teknik Pengujian

Pengujian perangkat lunak memerlukan perancangan kasus uji (*test case*) agar dapat menemukan kesalahan dalam waktu singkat dan usaha minimum. Berbagai macam metode perancangan kasus uji telah berevolusi. Metode-metode ini menyediakan *developer* pendekatan sistematis untuk pengujian. Terlebih lagi metode-metode ini menyediakan mekanisme yang dapat membantu memastikan kelengkapan dari pengujian dan menyediakan kemungkinan tertinggi untuk menemukan kesalahan-kesalahan dalam perangkat lunak [PRE-10:443]. Teknik atau metode perancangan kasus uji yang digunakan adalah *black-box testing*.

### 2.9.1.1 Black-Box Testing

*Black-box testing* atau *behavioral testing* berfokus pada persyaratan fungsional perangkat lunak [PRE-10:459]. 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.

### 2.9.2 Strategi Pengujian

Strategi untuk pengujian perangkat lunak mengintegrasikan metode desain *test case* perangkat lunak ke dalam sederetan langkah yang direncanakan dengan baik, dan hasilnya adalah konstruksi perangkat lunak yang berhasil [PRE-10:477]. Sejumlah strategi pengujian perangkat lunak telah diusulkan di dalam literatur. Strategi pengujian harus mengakomodasi pengujian tingkat rendah yang diperlukan untuk membuktikan bahwa segmen kode sumber yang kecil telah diimplementasikan dengan tepat, demikian juga pengujian tingkat tinggi yang memvalidasi fungsi-fungsi sistem mayor yang berlawanan dengan kebutuhan pelanggan. Proses pengujian dimulai dengan pengujian yang berfokus pada setiap modul secara individual (*unit testing*), dilanjutkan dengan pengujian integrasi

(*integration testing*) dan berakhir pada pengujian validasi (*validation testing*) [PRE-10:481].

### 2.9.2.1 PENGUJIAN VALIDASI

Validasi dapat ditentukan dengan berbagai cara, tetapi definisi yang sederhana adalah bahwa validasi berhasil bila sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan. 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-10:495].

### 2.9.2.2 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 *auto-reply* dari perangkat lunak SMS Gateway. 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-10:498].

### 2.9.2.3 PENGUJIAN USABILITY

Pengujian yang terakhir yaitu pengujian *usability*. Pengujian *usability* adalah pengujian terhadap atribut kualitas yang menilai seberapa mudahnya user interface yang digunakan dalam pengembangan suatu perangkat lunak [JAK-03:1]. Teknik kuisisioner dilakukan untuk mendukung pengujian ini. Sampel yang digunakan dalam pengujian ini sebanyak 20 sampel. Pertanyaan yang diajukan dalam kuisisioner ini didapatkan dari pihak yang sebelumnya sudah melakukan kuisisioner terhadap perangkat lunak berbasis web dengan pengujian *usability* yang sejenis. Pertanyaan – pertanyaan tersebut akan disertakan sebagai lampiran. Pertanyaan kuisisioner yang digunakan mengambil referensi dari web [BAL-13], karena pertanyaan kuisisioner dari halaman web tersebut dirasa sudah sesuai dan relevan dengan parameter pengujian *usability* yang akan dilakukan dalam skripsi ini.



## BAB III

### METODOLOGI PENELITIAN

#### 3.1 METODE PENELITIAN

Metode penelitian skripsi ini dilakukan dalam enam tahap yaitu studi literatur, analisa kebutuhan, perancangan sistem, implementasi, pengujian dan analisis, dan pengambilan kesimpulan.

##### 3.1.1 Studi literatur

Studi literatur menjelaskan kajian pustaka dan dasar teori yang digunakan untuk menunjang penulisan skripsi. Teori-teori pendukung tersebut meliputi:

- a. PHP adalah *server-side web scripting*, yaitu bahasa singkat (skrip) yang dikembangkan untuk aplikasi *web* yang dinamis
- b. *Database* adalah sekumpulan data, yang secara khas menggambarkan aktivitas satu atau lebih organisasi yang saling berhubungan.
- c. Melakukan observasi pemesanan tiket bus *online* dengan mengamati fitur dan layanan pemesanan tiket bus yang belum diaplikasikan pada web pemesanan tiket *online* lainnya untuk diterapkan pada web pemesanan tiket bus yang saya buat.
- d. SMS Gateway merupakan sebuah perangkat yang menawarkan layanan transit SMS, mentransformasikan pesan ke jaringan selular dari media lain, atau sebaliknya, sehingga memungkinkan pengiriman atau penerimaan pesan SMS dengan atau tanpa menggunakan ponsel.

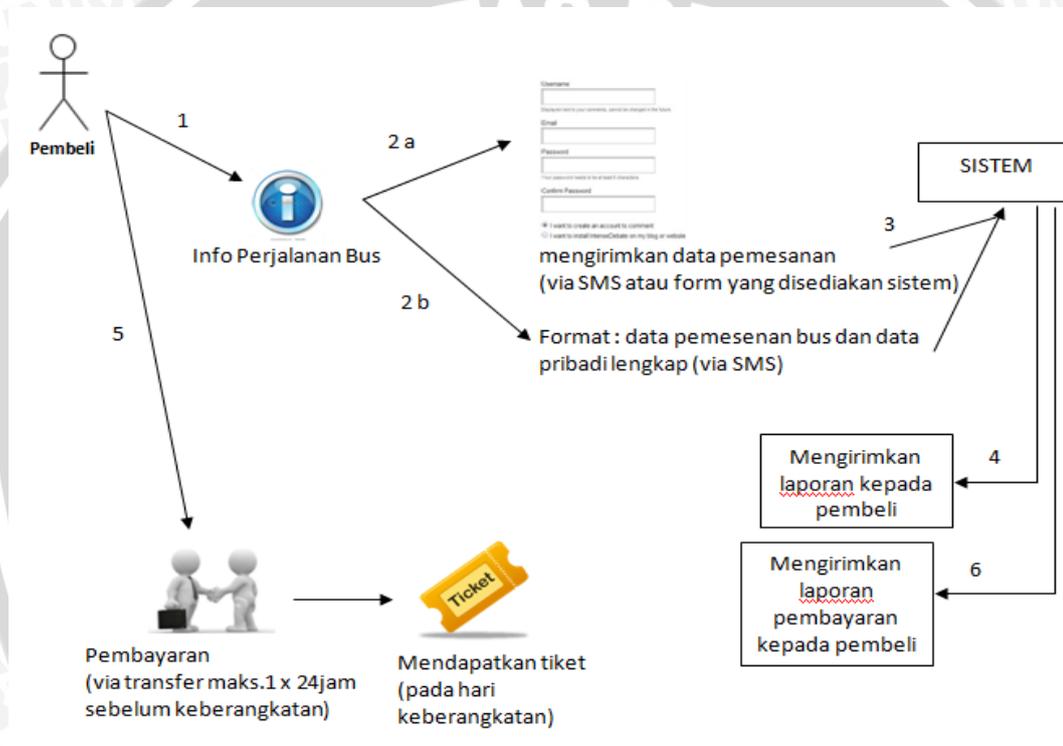
##### 3.1.2 Analisis kebutuhan

Analisa kebutuhan bertujuan untuk mendapatkan semua kebutuhan yang diperlukan dari sistem yang akan dibangun. Metode analisis yang digunakan adalah *Object Oriented Analysis* dengan menggunakan bahasa pemodelan UML (*Unified Modeling Language*). Diagram *Use Case* digunakan untuk mendeskripsikan kebutuhan-kebutuhan dan fungsionalitas sistem dari perspektif

end-user. Analisis kebutuhan dilakukan dengan mengidentifikasi semua kebutuhan (*requirements*) sistem.

### 3.1.3 perancangan sistem

Perancangan sistem dilakukan setelah semua kebutuhan sistem didapatkan melalui tahap analisis kebutuhan. Perancangan sistem berdasarkan *Object Oriented Analysis* dan *Object Oriented Design* yaitu menggunakan pemodelan UML (*Unified Modeling Language*). Sebagai gambaran awal, alur penggunaan sistem dapat dilihat pada gambar 3.1



Gambar 3.1 Diagram Blok Sistem  
Sumber : Perancangan

Pertama pengguna dapat melihat jadwal keberangkatan bus dan layanan yang tersedia pada web, kemudian pengguna mengirimkan data pemesanan melalui form pemesanan maupun melalui sms dengan format yang telah ditentukan. Sistem akan mengkonfirmasi data pemesanan pengguna apakah tiket tersedia atau tidak. Jika syarat-syarat data pemesanan telah terpenuhi, pengguna wajib melakukan pembayaran dengan batas waktu yang telah ditentukan. Jika pembayaran telah dipenuhi, pengguna mengirimkan bukti pembayaran kepada

sistem yang nantinya akan dikonfirmasi kembali oleh sistem. Pengguna bisa mendapatkan tiket dengan menunjukkan bukti konfirmasi pembelian melalui sms atau email.

### **3.1.4 implementasi**

Dalam implementasi terdapat proses pengkodean. Proses pengkodean atau *coding* dimaksudkan untuk merealisasikan desain yang telah dibuat secara bertahap dengan memperhatikan alur skenario-skenario yang telah dibuat. Implementasi perangkat lunak dilakukan dengan menggunakan bahasa pemrograman web yaitu PHP.

### **3.1.5 pengujian dan analisis**

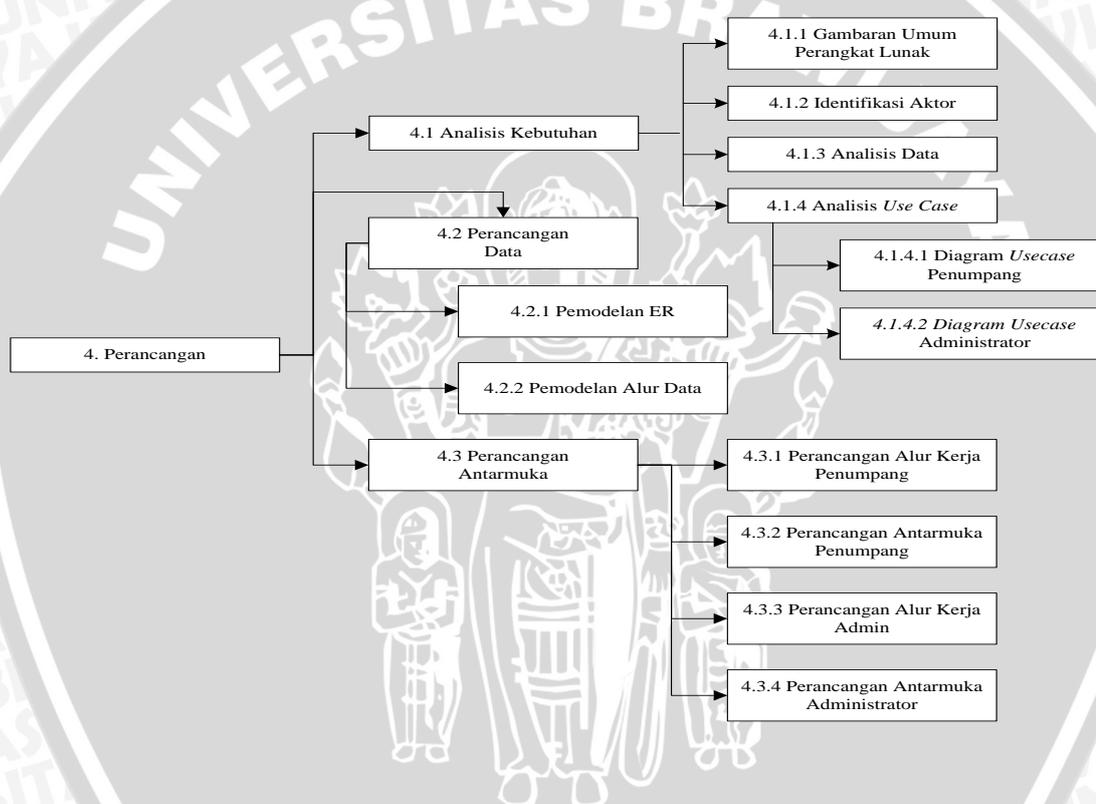
Pengujian aplikasi dilakukan untuk mengetahui kesesuaian analisis kebutuhan yang dibuat dengan implementasi aplikasi. Proses pengujian dilakukan tahapan-tahapan yaitu pengujian unit dan pengujian integrasi. Selain itu, dilakukan juga evaluasi sistem terhadap kepuasan konsumen dengan memberikan kuisisioner yang dapat diisi langsung oleh konsumen pada aplikasi tersebut. Analisis dari pengujian sistem dilakukan dengan membandingkan hasil-hasil evaluasi dengan persyaratan yang seharusnya dipenuhi oleh sistem.

### **3.1.6 pengambilan kesimpulan**

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

## BAB IV PERANCANGAN

Bab ini membahas mengenai perancangan perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang. Perancangan yang dilakukan meliputi dua tahap. Proses analisis kebutuhan dilakukan pada tahap pertama dan proses perancangan perangkat lunak dilakukan pada tahap kedua. Tahap-tahap perancangan yang dilakukan seperti yang digambarkan pada Gambar 4.1 berikut ini.



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

Tahap analisis kebutuhan terdiri atas empat langkah tahap. Tahap pertama melakukan penjabaran tentang gambaran umum perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang. Tahap kedua melakukan proses identifikasi aktor yang terlibat dalam sistem perangkat lunak. Tahap ketiga melakukan proses analisis data yang diperlukan. Tahap keempat melakukan analisis *use case* berdasarkan penjabaran gambaran umum perangkat lunak.

Tahap perancangan data memiliki 2 tahap yaitu pemodelan dengan ER dan pemodelan Alur Data.

Tahap perancangan antarmuka memiliki 2 tahap yaitu perancangan antarmuka penumpang dan perancangan antarmuka admin. Tiap tahap disertai dengan perancangan alur kerjanya masing-masing.

#### 4.1 ANALISIS KEBUTUHAN

Proses analisis kebutuhan mengacu pada gambaran umum perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang dan hasil pengumpulan, pemahaman dan penetapan kebutuhan – kebutuhan yang ingin didapatkan oleh pengguna. Proses analisis kebutuhan ini diawali dengan penjabaran gambaran umum perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang, identifikasi aktor yang terlibat, analisis data yang akan disimpan, penjabaran tentang daftar kebutuhan dan kemudian memodelkannya ke dalam diagram *data flow*. Analisis kebutuhan ini bertujuan untuk menggambarkan kebutuhan – kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna.

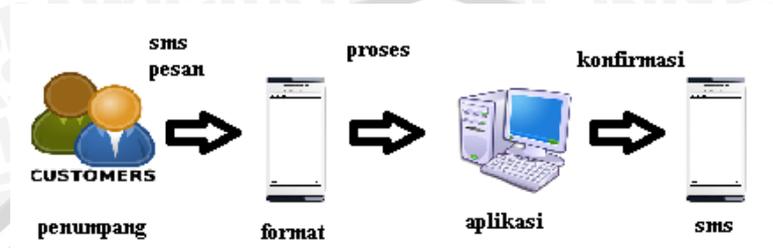
##### 4.1.1 Gambaran Umum Perangkat Lunak

Perangkat lunak ini adalah perangkat lunak yang berfungsi membantu sistem reservasi dengan menggunakan *sms gateway*. Sebelumnya sistem pemesanan di Terminal Arjosari Malang yang digunakan adalah sistem pemesanan manual, yang mengharuskan penumpang untuk datang ke terminal untuk pesan tiket. Arsitektur sistem awal reservasi sebelum aplikasi diterapkan akan ditunjukkan pada Gambar 4.2.



Gambar 4.2 Sistem Pemesanan Saat Ini  
Sumber : [Perancangan]

Skenario pemesanan saat ini adalah secara manual. Tiket dapat diterima dengan memesan langsung ke Terminal Arjosari Malang.. Arsitektur sistem yang akan dirancang ditunjukkan dalam Gambar 4.3.



**Gambar 4.3 Sistem Pemesanan Berbasis SMS GATEWAY**  
Sumber : [Perancangan]

Pada sistem pemesanan berbasis SMS *Gateway* yang diusulkan, skenario sistem adalah sebagai berikut. Penumpang mengirimkan SMS pemesanan tiket ke sistem sesuai dengan format yang telah ditentukan, kemudian sistem akan memeriksa apakah pesanan tiket penumpang tersedia atau tidak. Kemudian sistem akan mengirimkan SMS konfirmasi pemesanan kepada penumpang untuk melakukan pembayaran. Setelah melakukan pembayaran penumpang mengirim sms konfirmasi pembayaran ke sistem, lalu admin akan mengecek pembayaran tiket penumpang. Selanjutnya admin akan mengupdate status pembayaran tiket penumpang di sistem, kemudian sistem akan mengirim sms konfirmasi pembayaran penumpang. Penumpang cukup menunjukkan SMS konfirmasi pembayaran tiket kepada pegawai P.O bus yang dipesan untuk mendapatkan tiket.

Perangkat lunak ini memiliki 2 fungsi utama yaitu, aplikasi penumpang dan aplikasi admin.

a. Aplikasi Penumpang

Aplikasi dari perangkat lunak ini adalah aplikasi layanan SMS Gateway yang dapat diakses pengguna secara langsung melalui handphone dengan format SMS tertentu. Fungsi utama yang dapat dilakukan oleh aplikasi penumpang adalah :

1. Melihat info bus

Operasi ini bertujuan untuk melihat info bus dari sisi penumpang melalui SMS.

2. Memesan tiket

Operasi ini bertujuan untuk melakukan pemesanan tiket dari sisi penumpang melalui SMS.

3. Mengkonfirmasi pembayaran

Operasi ini bertujuan untuk melakukan konfirmasi pembayaran pesanan tiket dari sisi penumpang melalui SMS.

- b. Aplikasi Admin

Aplikasi admin dari perangkat lunak ini adalah aplikasi yang digunakan untuk mengelola elemen – elemen perangkat lunak secara lebih lanjut. Proses – proses utama yang terdapat dalam aplikasi admin adalah sebagai berikut :

1. Login

Operasi ini bertujuan untuk memberi seleksi kepada pengguna sehingga hanya admin saja yang dapat mengelola sistem.

2. Mengelola Data Pengemudi

Operasi ini bertujuan untuk melakukan manipulasi data pengemudi.

Operasi ini terdiri atas beberapa operasi, yaitu :

- Melakukan tambah data pengemudi.
- Melakukan edit data pengemudi.
- Melakukan *update* data pengemudi.
- Melakukan *delete* data pengemudi.

3. Mengelola Data Bus

Operasi ini bertujuan untuk melakukan manipulasi data bus. Operasi ini terdiri atas beberapa operasi, yaitu :

- Melakukan tambah data bus.
- Melakukan edit pada data bus.
- Melakukan *update* data bus.
- Melakukan *delete* data bus.
- Melakukan *view* data bus.

#### 4. Mengelola Jadwal Bus

Operasi ini bertujuan untuk melakukan manipulasi data jadwal bus.

Operasi ini terdiri atas beberapa operasi, yaitu :

- Melakukan tambah jadwal pada data jadwal bus.
- Melakukan tambah penjadwalan pada data bus.
- Melakukan *view* pada data bus.

#### 5. Mengelola Data Pemesanan

Operasi ini bertujuan untuk melakukan konfirmasi terhadap pesanan tiket calon penumpang. Operasi ini terdiri atas beberapa operasi, yaitu

- Melakukan konfirmasi pada data pemesanan.
- Melakukan *update* data pemesanan.

#### 6. Mengelola Data Admin.

Operasi ini bertujuan untuk melakukan penambahan data admin.

#### 7. Logout.

Operasi *Logout* dilakukan bila pengguna yang terdaftar sebagai Admin telah selesai menggunakan aplikasi ini.

### 4.1.2 Identifikasi Aktor

Tahap ini adalah tahap untuk melakukan identifikasi terhadap aktor - aktor yang akan berinteraksi dengan perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang. Tabel 4.1 memperlihatkan aktor – aktor yang terlibat beserta penjelasannya masing-masing yang merupakan hasil dari proses identifikasi aktor.

**Tabel 4.1 Identifikasi aktor**

<b>Aktor</b>	<b>Deskripsi</b>
<b>Penumpang</b>	Penumpang adalah pengguna biasa yang bisa memesan tiket bus melalui sms atau web.
<b>Admin</b>	Admin adalah pengguna aplikasi admin perangkat lunak Sistem Reservasi Tiket Bus Menggunakan PHP dan SMS GATEWAY dan dapat mengelola seluruh data lebih lanjut.

Sumber : [Perancangan]

### 4.1.3 Analisis Data

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

1. Data Bus yang terdiri dari kode bus, no jadwal, id pengemudi nama bus, gambar bus, harga. Data ini berfungsi menyimpan data bus yang ada di Terminal Arjosari.
2. Data Jadwal yang terdiri dari no jadwal, tujuan, tanggal berangkat, jam berangkat, jam tiba. Data ini berfungsi menyimpan data jadwal keberangkatan dan kedatangan bus di Terminal Arjosari.
3. Data Pemesanan yang terdiri dari no pesan, tanggal pesan, kode bus, nama, alamat, email, telepon, jumlah tiket, via, status konfirmasi, status bayar. Data ini berfungsi menyimpan data pemesanan tiket penumpang.
4. Data pengemudi yang terdiri dari id pengemudi, kode bus, nama pengemudi, alamat pengemudi, no telepon. Data ini berfungsi menyimpan data pengemudi bus yang berada di Terminal Arjosari.
5. Data admin yang terdiri dari name dan password. Data ini berfungsi untuk menyimpan data admin.

### 4.1.4 Analisis Use Case

Analisis *use case* dibagi menjadi 2, yaitu daftar kebutuhan dan diagram *use case*. Pada daftar kebutuhan akan dispesifikasikan menjadi 2 yaitu spesifikasi kebutuhan penumpang dan spesifikasi kebutuhan admin. Spesifikasi kebutuhan penumpang ditunjukkan pada Tabel 4.2.

Tabel 4.2 Spesifikasi kebutuhan penumpang

Nomor SRS	Kebutuhan	Use Case
SRS_001_01	Perangkat lunak harus mampu membaca SMS yang diterima oleh sistem SMS GATEWAY untuk melihat info bus	Melihat Info Bus via SMS
SRS_001_02	Perangkat lunak harus mampu menyediakan info bus yang dicari via <i>website</i>	Melihat Info Bus via <i>Website</i>

SRS_001_03	Perangkat lunak harus mampu membaca SMS yang diterima oleh sistem SMS GATEWAY untuk mengkonfirmasi pemesanan tiket via SMS	Mengkonfirmasi pemesanan tiket via SMS
SRS_001_04	Perangkat lunak harus mampu mengkonfirmasi pemesanan tiket yang dilakukan via <i>website</i>	Mengkonfirmasi pesanan tiket via <i>Website</i>
SRS_001_05	Perangkat lunak harus mampu membaca SMS yang diterima oleh sistem SMS GATEWAY untuk mengkonfirmasi pembayaran .	Mengkonfirmasi pembayaran via SMS

Sumber : Perancangan

Spesifikasi kebutuhan admin ditunjukkan pada tabel 4.3.

**Tabel 4.3 Spesifikasi kebutuhan admin**

Nomor SRS	Kebutuhan	Use Case
SRS_002_01	Perangkat lunak harus menyediakan fasilitas untuk login sehingga hanya admin saja yang dapat mengelola sistem.	<i>Login</i>
SRS_002_02	Perangkat lunak harus menyediakan fasilitas untuk menjalankan SMS GATEWAY.	Menjalankan SMS GATEWAY
SRS_002_03	Perangkat lunak harus menyediakan fasilitas untuk mengelola data pengemudi.	Mengelola data pengemudi
SRS_002_04	Perangkat lunak harus menyediakan fasilitas untuk mengelola data bus.	Mengelola data bus
SRS_002_05	Perangkat lunak harus menyediakan fasilitas untuk mengelola data jadwal bus.	Mengelola data jadwal bus
SRS_002_06	Perangkat lunak harus menyediakan fasilitas untuk mengelola data pesanan tiket .	Mengelola data pemesanan
SRS_002_07	Perangkat lunak harus menyediakan fasilitas untuk mengelola data admin.	Mengelola data <i>users</i>
SRS_002_08	Perangkat lunak harus menyediakan fasilitas <i>logout</i> sehingga admin dapat meninggalkan aplikasi sistem.	<i>Logout</i>

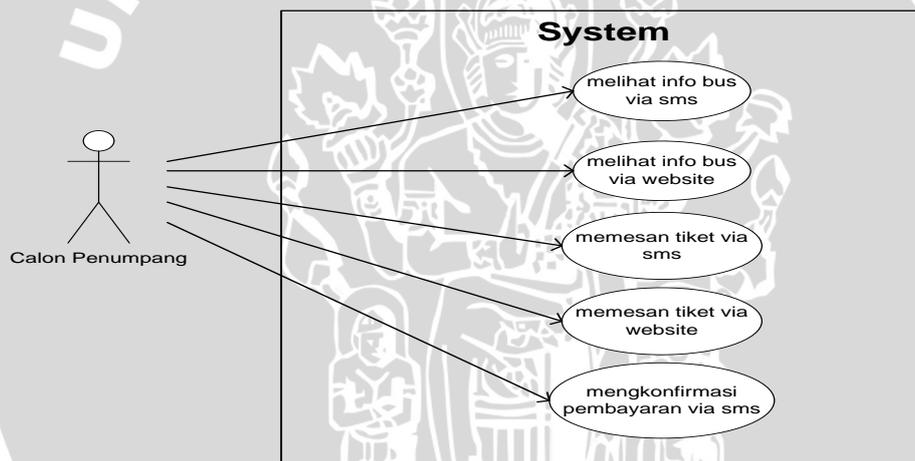
Sumber : Perancangan

Diagram *use case* adalah salah satu diagram untuk memodelkan aspek perilaku sistem. Diagram *use case* menunjukkan sekumpulan *use case*, aktor dan hubungannya. *Use case* merupakan fungsionalitas dari sistem yang diinisialisasi oleh aktor. Pemodelan diagram *use case* yang menggambarkan fungsionalitas

perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang dibagi menjadi dua yaitu diagram *use case* untuk aplikasi dan diagram *use case* untuk aplikasi admin.

#### 4.1.4.1 Diagram Use Case Penumpang

Diagram *use case* ini melibatkan Penumpang sebagai aktor dan empat buah *use case*. Empat buah *use case* ini termasuk dalam bagian *software* SMS Gateway pada sistem reservasi tiket Terminal Arjosari Malang. Empat buah *use case* ini juga akan disertai dengan *scenario use case* untuk menjelaskan rangkaian aktifitas yang terjadi di *use case* tersebut. Diagram *use case* untuk aplikasi ditunjukkan pada gambar 4.4 berikut ini :



Gambar 4.4 Diagram Use Case Penumpang  
Sumber : Perancangan

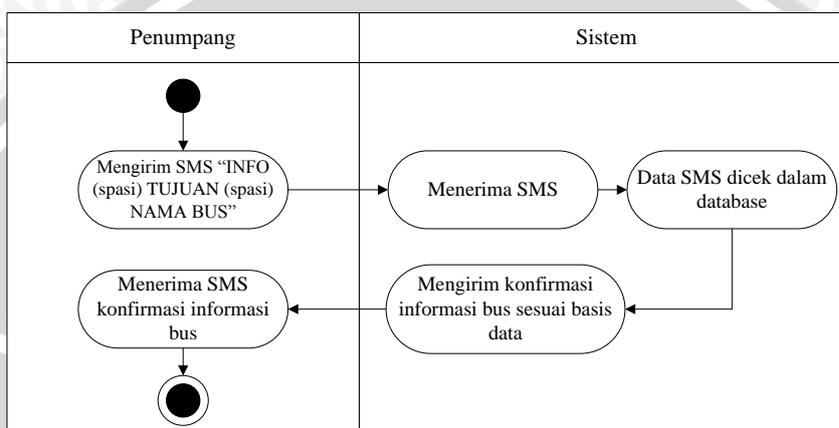
#### 1. Skenario Use Case Melihat Info Bus Melalui SMS

Tabel 4.4 Use case Melihat info bus Melalui SMS

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_01
Nama	Melihat info bus Melalui SMS
Tujuan	Untuk melihat info melalui SMS Gateway.
Deskripsi	Use Case ini menjelaskan bagaimana melakukan proses melihat info bus melalui SMS Gateway.
Aktor	Penumpang
Skenario Utama	
Kondisi Awal	Perangkat lunak SMS Gateway Informasi Bus sudah berjalan dan akses GAMMU sudah terkoneksi dengan database bus.
Aksi Aktor	Reaksi Sistem

Penumpang SMS ke nomor server dengan format “INFO (spasi) NAMA BUS(spasi) KOTA TUJUAN (spasi) TANGGAL BERANGKAT”	SMS diterima oleh sistem. Sistem akan mengecek data ke database. Kemudian sistem akan mengirim SMS balasan kepada penumpang.
Kondisi Akhir	Penumpang mendapatkan SMS Konfirmasi Info dari sistem berupa “Informasi Bus [nama_bus] [tujuan] [tgl_berangkat] [jam berangkat] [harga]. Cek www.tiketbusonline.com.”

Sumber : Perancangan



Gambar 4.5 Diagram aktifitas melihat info bus melalui SMS

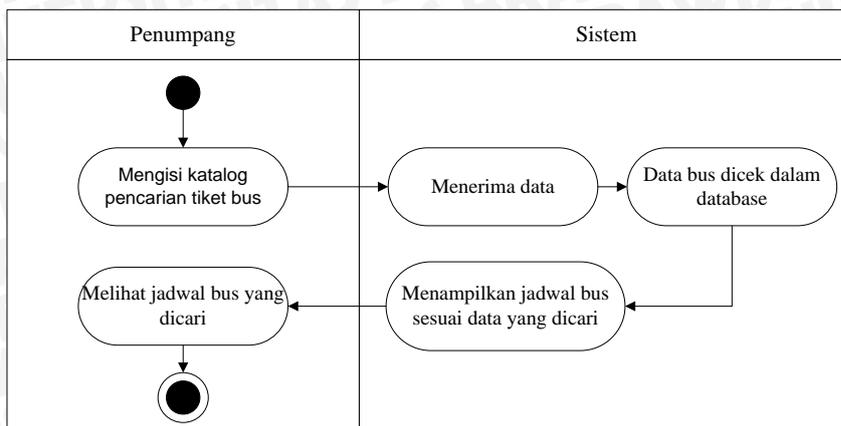
Sumber : Perancangan

## 2. Skenario Use Case Melihat Info Bus Melalui Website

Tabel 4.5 Use case Melihat info bus Melalui Website

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_02
Nama	Melihat info bus Melalui Website
Tujuan	Untuk melihat info melalui Website.
Deskripsi	Use Case ini menjelaskan bagaimana melakukan proses melihat info bus melalui Website.
Aktor	Calon Penumpang
Skenario Utama	
Kondisi Awal	Sistem menampilkan kolom pencarian tiket bus.
Aksi Aktor	Reaksi Sistem
Penumpang mengisi kolom pencarian tiket bus dengan tanggal keberangkatan dan tujuan yang diinginkan	Data diterima oleh sistem dan dicek di database. Apabila data yang dicari tersedia maka jadwal akan di tampilkan di halaman website. Apabila data yang dicari tidak tersedia sistem akan menampilkan jadwal yang dicari tidak tersedia di halaman website.
Kondisi Akhir	Sistem menampilkan data bus yang dicari dengan keterangan nama bus, tujuan, jam berangkat, jam tiba, tarif dan aksi di halaman website.

Sumber : Perancangan



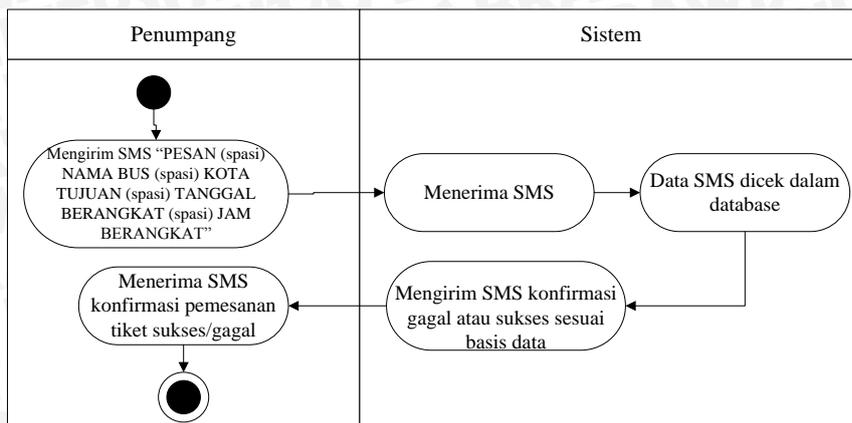
Gambar 4.6 Diagram aktifitas melihat info bus melalui Website  
 Sumber : Perancangan

### 3. Skenario Use Case Memesan Tiket Melalui SMS

Tabel 4.6 Use case memesan tiket melalui SMS

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_02
Nama	Memesan Tiket Melalui SMS
Tujuan	Untuk memesan tiket melalui SMS Gateway.
Deskripsi	Use Case ini menjelaskan bagaimana melakukan proses pemesanan tiket melalui SMS Gateway.
Aktor	Penumpang
Skenario Utama	
Kondisi Awal	Perangkat lunak SMS Gateway Reservasi Tiket sudah berjalan dan akses GAMMU sudah terkoneksi dengan database bus.
Aksi Aktor	Reaksi Sistem
SMS ke nomor server dengan format "PESAN (spasi) NAMA BUS (spasi) KOTA TUJUAN (spasi) TANGGAL BERANGKAT (spasi) JUMLAH TIKET"	SMS diterima oleh sistem. Sistem akan mengecek data ke database. Kemudian sistem akan mengirim SMS balasan kepada penumpang.
Kondisi Akhir	Penumpang mendapatkan SMS Konfirmasi Info dari sistem berupa "Pesanan Anda Dengan [no_order] [total_bayar]. Tunggu Konfirmasi Selanjutnya."

Sumber : Perancangan



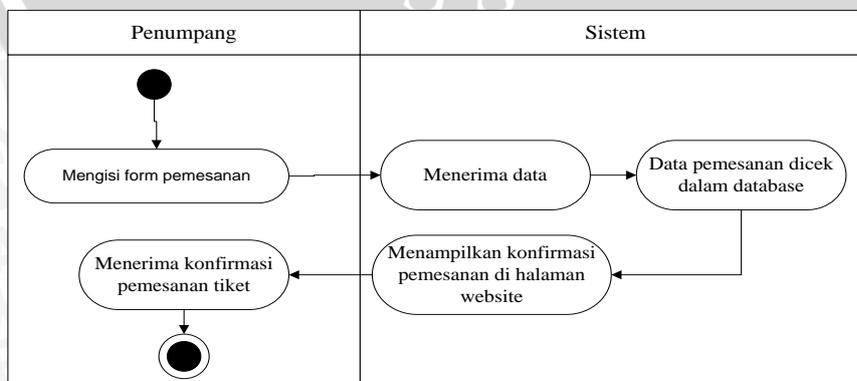
Gambar 4.7 Diagram aktifitas memesan tiket melalui SMS  
 Sumber : Perancangan

#### 4. Skenario Use Case Memesan Tiket Melalui Website

Tabel 4.7 Use case memesan tiket melalui Website

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_03
Nama	Memesan Tiket Melalui Website
Tujuan	Untuk memesan tiket melalui Website.
Deskripsi	Use Case ini menjelaskan bagaimana melakukan proses pemesanan tiket melalui Website.
Aktor	Penumpang
Skenario Utama	
Kondisi Awal	Sistem menampilkan form pemesanan.
Aksi Aktor	Reaksi Sistem
Mengisi form pemesanan yang telah disediakan oleh sistem	Data diterima oleh sistem. Apabila format data telah sesuai dengan kriteria sistem, sistem akan menampilkan konfirmasi pemesanan di halaman website.
Kondisi Akhir	Sistem menampilkan konfirmasi pemesanan tiket di halaman website berupa Tiket berhasil anda pesan, segera lakukan pembayaran dengan melalu rekening di bawah ini. No Rekening : BCA "[no rekening]". Terminal Arjosari.

Sumber : Perancangan



Gambar 4.8 Diagram aktifitas memesan tiket melalui Website  
 Sumber : Perancangan

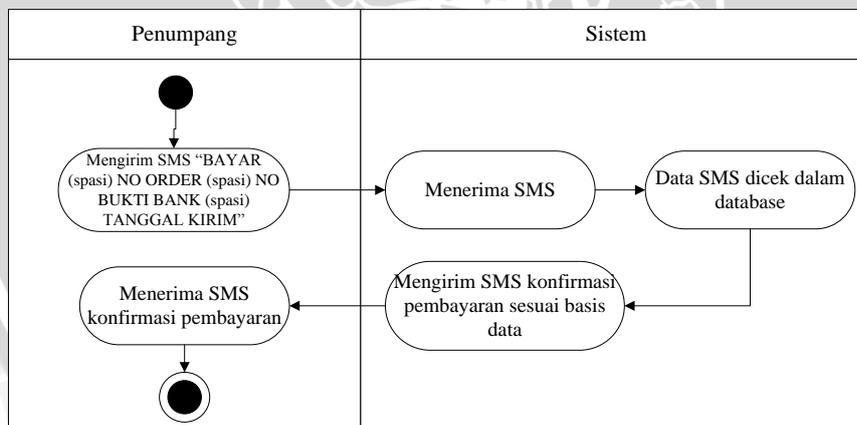


## 5. Skenario Use Case Mengkonfirmasi Pembayaran Melalui SMS

Tabel 4.8 Use case mengkonfirmasi pembayaran Melalui SMS

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_001_04
Nama	Mengkonfirmasi pembayaran Melalui SMS
Tujuan	Untuk mengkonfirmasi pembayaran melalui SMS Gateway.
Deskripsi	Use Case ini menjelaskan bagaimana melakukan proses mengkonfirmasi pembayaran melalui SMS Gateway.
Aktor	Penumpang
Skenario Utama	
Kondisi Awal	Perangkat lunak SMS Gateway Reservasi Tiket sudah berjalan dan akses GAMMU sudah terkoneksi dengan database pemesanan.
Aksi Aktor	Reaksi Sistem
SMS ke nomor server dengan format “BAYAR (spasi) NO ORDER (spasi) NO BUKTI BANK (spasi) TANGGAL KIRIM”	SMS diterima oleh sistem. Apabila format SMS telah sesuai dengan kriteria sistem, maka admin akan mengecek ke database pemesanan. Admin akan mengupdate status konfirmasi pembayaran. Kemudian sistem akan mengirim SMS konfirmasi pembayaran kepada penumpang.
Kondisi Akhir	Penumpang mendapatkan SMS Konfirmasi Pesanan Tiket dari sistem berupa “Terima kasih atas pembayaran dari pemesanan anda dengan [no.order] [nama_bus] [tujuan] [tanggal berangkat].”

Sumber : Perancangan

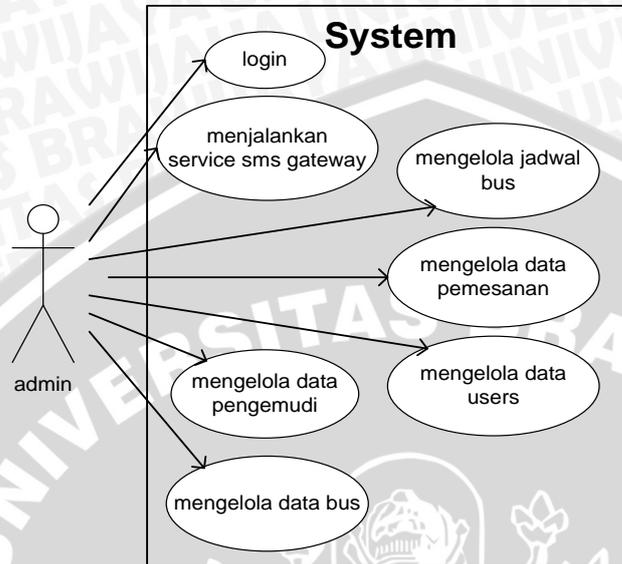


Gambar 4.9 Diagram aktifitas mengkonfirmasi pembayaran melalui SMS  
Sumber : Perancangan

### 4.1.4.2 Diagram Use Case Admin

Diagram *use case* ini melibatkan admin sebagai aktor dan 6 buah *use case*. 6 buah *use case* ini termasuk dalam bagian *software* SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang. 6 buah *use case* ini juga akan

disertai dengan skenario *use case* untuk menjelaskan rangkaian aktifitas yang terjadi di masing - masing *use case* tersebut. Diagram *use case* untuk aplikasi admin ditunjukkan pada Gambar 4.10 berikut ini :



**Gambar 4.10 Diagram *use case* admin**  
**Sumber : Perancangan**

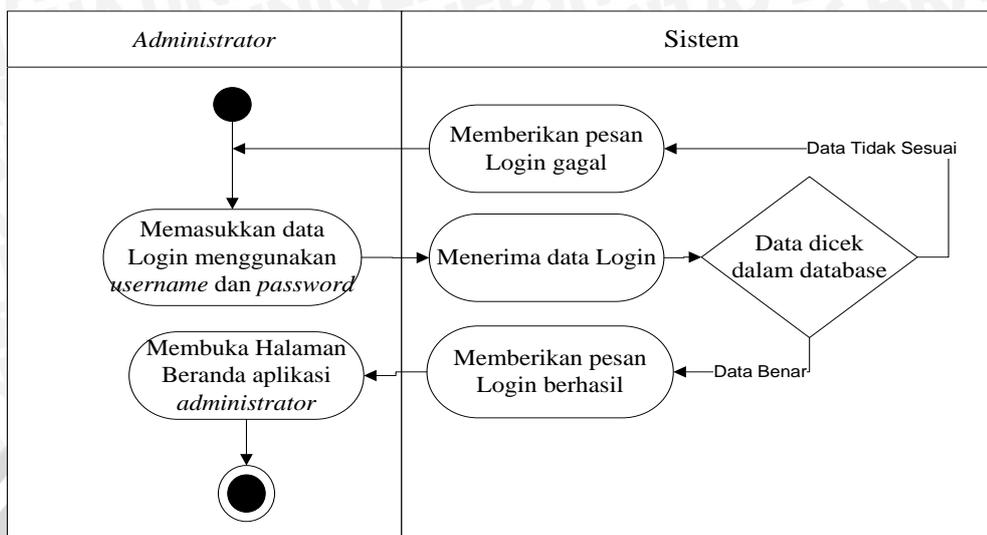
### 1. Skenario Use Case Login

**Tabel 4.9 Use case Login**

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_01
Nama	Login
Tujuan	Untuk menyeleksi admin yang sah.
Deskripsi	Use case ini menjelaskan bagaimana admin melakukan login untuk dapat menampilkan halaman utama sisi admin.
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan form login.
Aksi Aktor	Reaksi Sistem
Admin memasukkan data login ( <i>name</i> dan <i>password</i> ), lalu menekan tombol Login.	Sistem menerima data login dan kemudian sistem melakukan pengecekan terhadap data yang dimasukkan oleh admin. Jika data benar, maka akan ditampilkan halaman utama admin.
Skenario Alternatif 1 : Jika <i>name</i> atau <i>password</i> kosong	
	1. Sistem akan menampilkan pesan peringatan bahwa <i>name</i> atau <i>password</i> kosong.
Skenario Alternatif 2 : Jika <i>name</i> dan <i>password</i> salah	
	2. Sistem akan menampilkan pesan peringatan bahwa <i>name</i> dan <i>password</i> salah.

Kondisi Akhir	Sistem menampilkan halaman utama sisi admin.
---------------	----------------------------------------------

Sumber : Perancangan



Gambar 4.11 Diagram aktifitas login

Sumber : Perancangan

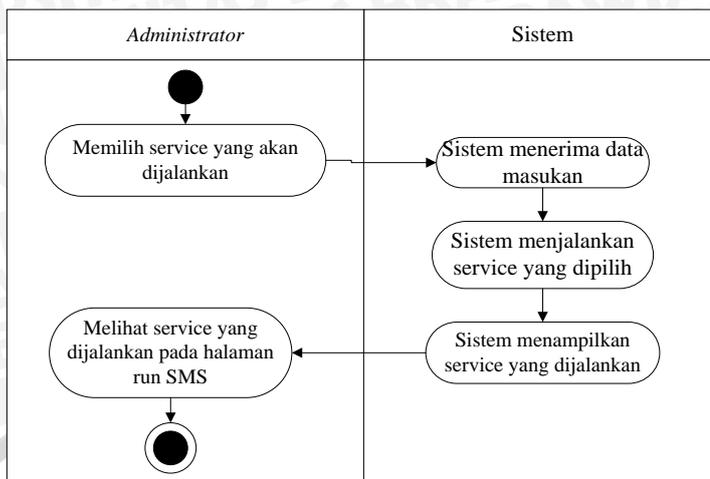
## 2. Skenario Use Case Menjalankan Service SMS GATEWAY

Tabel 4.10 Use case Menjalankan SMS GATEWAY

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_02
Nama	Menjalankan Service SMS GATEWAY
Tujuan	Untuk menjalankan service sms gateway.
Deskripsi	Use case ini menjelaskan bagaimana admin menjalankan service sms gateway
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan form service sms gateway.
Aksi Aktor	Reaksi Sistem
Admin memilih servis yang ingin dijalankan.	Sistem menjalankan service sesuai dengan service yang dipilih oleh admin.
Kondisi Akhir	Sistem menampilkan halaman service yang dipilih sisi admin.

Sumber : Perancangan





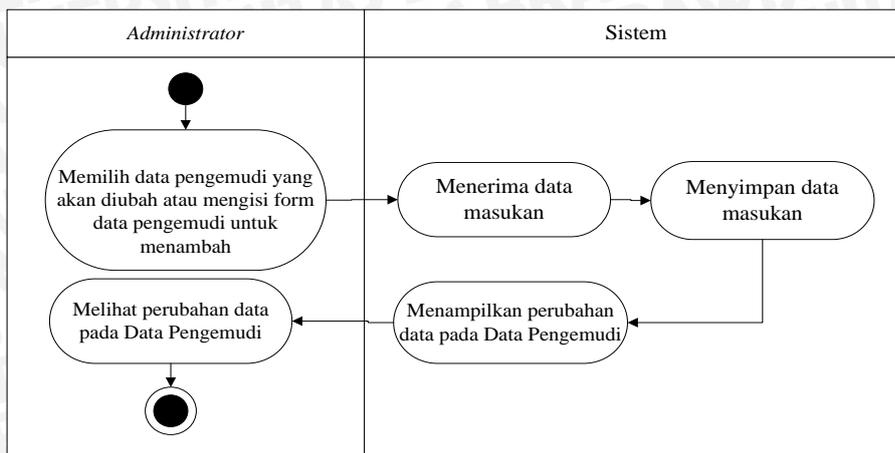
Gambar 4.12 Diagram aktifitas menjalankan Service SMS Gateway  
 Sumber : Perancangan

### 3. Skenario Use Case Mengelola Data pengemudi

Tabel 4.11 Use case Mengelola Data Pengemudi

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_03
Nama	Mengelola Data Pengemudi
Tujuan	Untuk melakukan pengolahan data pengemudi.
Deskripsi	Use case ini menjelaskan bagaimana admin dapat melakukan pengelolaan data pengemudi (tambah, ubah, atau hapus).
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan tabel data pengemudi.
Aksi Aktor	Reaksi Sistem
Admin memilih data dari daftar pengemudi sehingga muncul form tambah, ubah, atau hapus.	Sistem menerima data pengemudi yang akan ditambahkan atau diubah ke dalam database atau data id pengemudi yang akan dihapus, kemudian sistem melakukan proses penambahan data penghapusan data terhadap pengemudi yang dimasukkan oleh admin sesuai dengan tombol yang ditekan.
Kondisi Akhir	Data pengemudi yang berada di database berhasil ditambah, diubah, atau dihapus dan data pada daftar akan berubah sesuai dengan operasi yang dilakukan.

Sumber : Perancangan



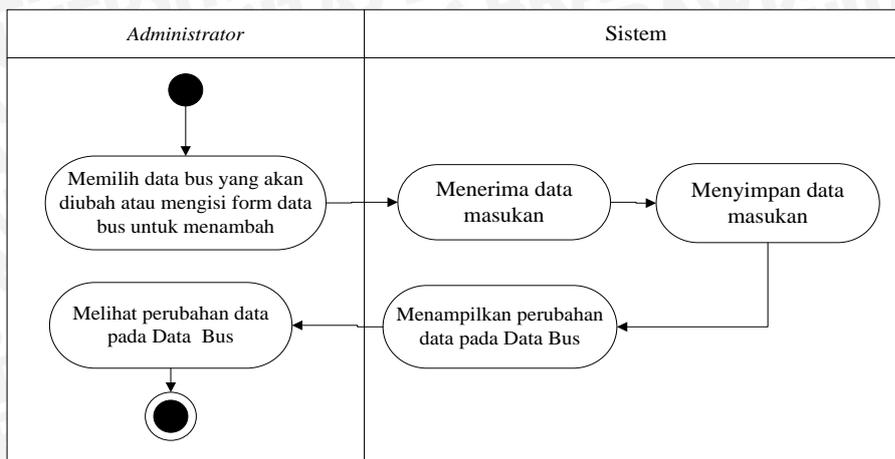
Gambar 4.13 Diagram aktifitas mengelola data pengemudi  
 Sumber : Perancangan

#### 4. Skenario Use Case Mengelola Data Bus

Tabel 4.12 Use case Mengelola Data Bus

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_04
Nama	Mengelola Data Bus
Tujuan	Untuk melakukan pengelolaan data bus.
Deskripsi	Use case ini menjelaskan bagaimana admin dapat melakukan pengelolaan data bus (tambah, ubah, atau hapus).
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan daftar dan form data bus.
Aksi Aktor	Reaksi Sistem
Admin memilih data dari daftar bus sehingga muncul form tambah, ubah, atau hapus.	Sistem menerima data bus yang akan ditambahkan atau diubah ke dalam database atau data id bus yang akan dihapus, kemudian sistem melakukan proses penambahan data penghapusan data terhadap bus yang dimasukkan oleh admin sesuai dengan tombol yang ditekan.
Kondisi Akhir	Data bus yang berada di database berhasil ditambah, diubah, atau dihapus dan data pada daftar akan berubah sesuai dengan operasi yang dilakukan.

Sumber :Perancangan



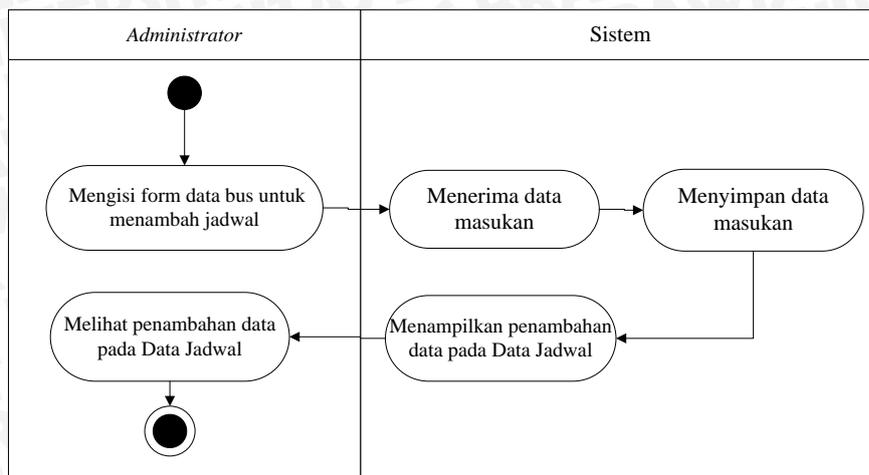
Gambar 4.14 Diagram aktifitas mengelola data bus  
 Sumber : Perancangan

## 5. Skenario Use Case Mengelola Data Jadwal Bus

Tabel 4.13 Use case Mengelola Data Jadwal Bus

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_003_05
Nama	Mengelola Data Jadwal Bus
Tujuan	Untuk melakukan pengelolaan data jadwal bus
Deskripsi	Use case ini menjelaskan bagaimana admin dapat melakukan pengelolaan data jadwal bus (Tambah Jadwal, Tambah Penjadwalan).
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan daftar jadwal bus.
Aksi Aktor	Reaksi Sistem
Admin menekan tombol tambah untuk menambahkan jadwal bus.	Sistem menerima data jadwal bus yang akan ditambahkan ke dalam <i>database</i> , kemudian sistem melakukan proses penambahan data terhadap data jadwal bus yang dimasukkan oleh admin sesuai dengan data yang dimasukkan.
Kondisi Akhir	Data jadwal bus yang berada di <i>database</i> berhasil ditambah dan data pada daftar akan berubah sesuai dengan operasi yang dilakukan.

Sumber : Perancangan



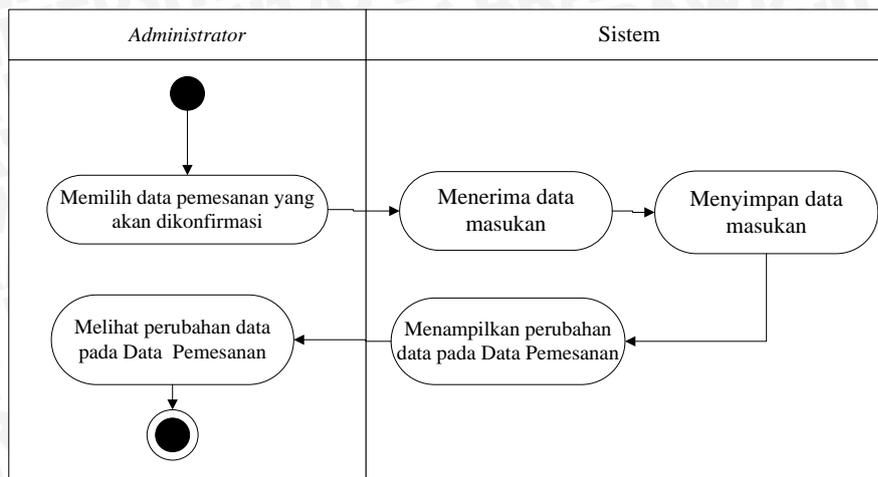
Gambar 4.15 Diagram aktifitas mengelola data jadwal bus  
 Sumber : Perancangan

## 6. Skenario Use Case Mengelola Data Pemesanan

Tabel 4.14 Use case Mengelola Data Pemesanan

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_06
Nama	Mengelola Data Pemesanan
Tujuan	Untuk melakukan pengelolaan data pemesanan.
Deskripsi	Use case ini menjelaskan bagaimana admin dapat melakukan pengelolaan data pemesanan (konfirmasi).
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan daftar dan form data pemesanan.
Aksi Aktor	Reaksi Sistem
Admin memilih data dari daftar pemesanan sehingga muncul pilhan status konfirmasi, dan menekan tombol Belum, Sudah atau Tolak.	Sistem menerima data pemesanan yang akan ditambahkan ke dalam database atau data id pemesanan yang akan dikonfirmasi, kemudian sistem melakukan proses perubahan status data terhadap data pemesanan yang dimasukkan oleh admin sesuai dengan option yang dipilih.
Kondisi Akhir	Data pemesanan yang berada di database berhasil dirubah statusnya dan data pada daftar akan berubah sesuai dengan operasi yang dilakukan.

Sumber : Perancangan



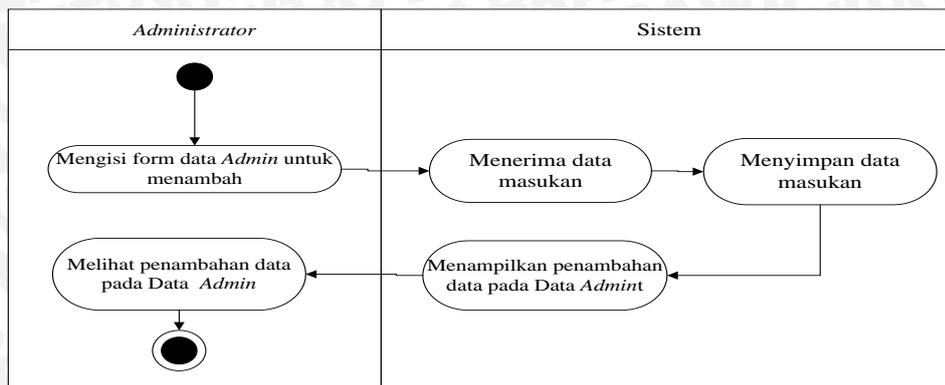
Gambar 4.16 Diagram aktifitas mengelola data pemesanan  
 Sumber : Perancangan

### 7. Skenario Use Case Mengelola Data Users

Tabel 4.15 Use case Mengelola Data Users

Skenario Kasus Pada Sistem	
Nomor Use Case	SRS_002_07
Nama	Mengelola Data Users
Tujuan	Untuk melakukan pengelolaan data users.
Deskripsi	Use case ini menjelaskan bagaimana admin dapat melakukan pengelolaan data users (tambah, ubah, atau hapus).
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan daftar dan form data users.
Aksi Aktor	Reaksi Sistem
Admin memilih data dari daftar users sehingga muncul form tambah, ubah, atau hapus.	Sistem menerima data users yang akan ditambahkan atau diubah ke dalam database atau data id users yang akan dihapus, kemudian sistem melakukan proses penambahan data penghapusan data terhadap data users yang dimasukkan oleh admin sesuai dengan tombol yang ditekan.
Kondisi Akhir	Data admin yang berada di database berhasil ditambah, diubah, atau dihapus dan data pada daftar akan berubah sesuai dengan operasi yang dilakukan.

Sumber : Perancangan



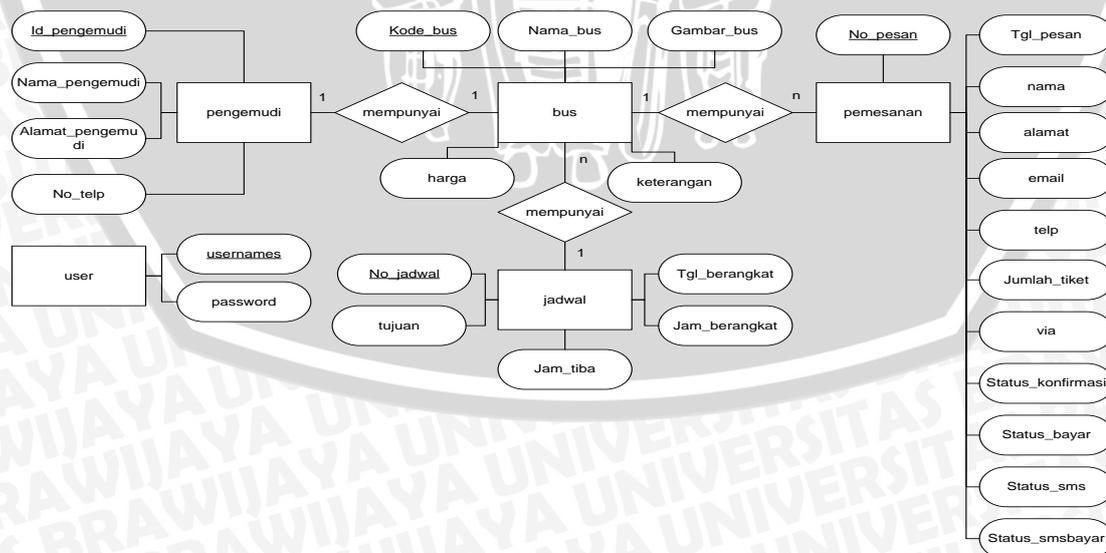
Gambar 4.17 Diagram aktifitas mengelola data admin  
 Sumber : Perancangan

## 4.2 PERANCANGAN DATA

Perancangan data memiliki dua tahap, yaitu pemodelan dengan ERD (Entity Relationship Diagram) dan Pemodelan dengan DFD.

### 4.2.1 Pemodelan Dengan ERD

Basis data berfungsi sebagai tempat menyimpan data. Pemodelan dengan ERD digunakan untuk merancang basis data yang akan dibuat agar masukan dan keluaran program sesuai dengan apa yang diharapkan. Pemodelan dengan ERD mengambil acuan dari proses analisis data yang dilakukan pada tahap analisis kebutuhan. Arsitektur basis data yang akan dirancang dijelaskan pada Gambar 4.18.



Gambar 4.18 Diagram *entity relationship* basis data  
 Sumber : [Perancangan]



Berikut ini merupakan struktur tabel serta keterangan masing masing tabel dan *field* yang ada pada *database*. Tabel berisi *names* dan *password* yang dapat dimiliki pengguna aplikasi SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang. Struktur tabel admin ditunjukkan pada Tabel 4.16.

**Tabel 4.16 Struktur tabel admin**

No.	Nama Field	Keterangan
1	Username	nama Admin
2	Password	Password Admin

Sumber : [Perancangan]

Tabel jadwal berisi data jadwal bus. Data jadwal tersebut meliputi nomor jadwal, tujuan, tanggal berangkat, jam berangkat, jam tiba. Struktur tabel jadwal ditunjukkan pada Tabel 4.17.

**Tabel 4.17. Struktur tabel jadwal**

No.	Nama Field	Keterangan
1	no_jadwal	No jadwal bus
2	Tujuan	Tujuan bus
3	tgl_berangkat	Tanggal keberangkatan bus
4	jam_berangkat	Jam keberangkatan bus
5	jam_tiba	Jam kedatangan bus

Sumber : [Perancangan]

Data pengemudi tersebut meliputi *id* pengemudi, kode bus, nama pengemudi, alamat pengemudi dan nomor telepon. Struktur tabel pengemudi ditunjukkan pada Tabel 4.18.

**Tabel 4.18. Struktur tabel pengemudi**

No.	Nama Field	Keterangan
1	id_pengemudi	Id pengemudi
2	kode_bus	Kode bus
3	nama_pengemudi	Nama pengemudi
4	alamat_pengemudi	Alamat pengemudi
5	no_telp	No telepon pengemudi

Sumber : [Perancangan]

Data bus tersebut meliputi kode bus, nomor jadwal, *id* pengemudi, nama bus, gambar bus, harga, keterangan. Struktur tabel bus ditunjukkan pada Tabel 4.19.

**Tabel 4.19. Struktur tabel bus**

No.	Nama Field	Keterangan
1	kode_bus	Kode bus
2	no_jadwal	No jadwal bus
3	id_pengemudi	Id pengemudi
4	nama_bus	Nama bus
5	gambar_bus	Gambar bus

6	Harga	Harga tiket bus
7	Keterangan	Keterangan

Sumber : [Perancangan]

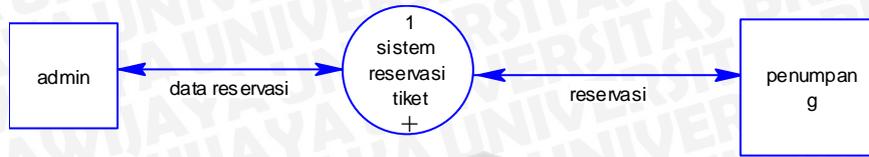
Data pemesanan meliputi nomor pesan, tanggal pesan, kode bus, nama, alamat, email, telepon, jumlah tiket, via, status konfirmasi, status bayar, status sms dan status smsbayar. Struktur pemesanan ditunjukkan pada Tabel 4.20.

**Tabel 4.20. Struktur tabel pesan**

No.	Nama Field	Keterangan
1	no_pesan	Nomor pemesanan
2	tgl_pesan	Tanggal pemesanan
3	kode_bus	Kode bus
4	Nama	Nama pemesan
5	Alamat	Alamat pemesan
6	Email	Email pemesan
7	Telp	Telepon pemesan
8	jumlah_tiket	Jumlah tiket pemesan
9	Via	Pemesanan via
10	status_konfirmasi	Status konfirmasi pemesanan
11	Status_bayar	Status pembayaran
12	Status_sms	Status sms
13	Status_smsbayar	Status konfirmasi sms pembayaran

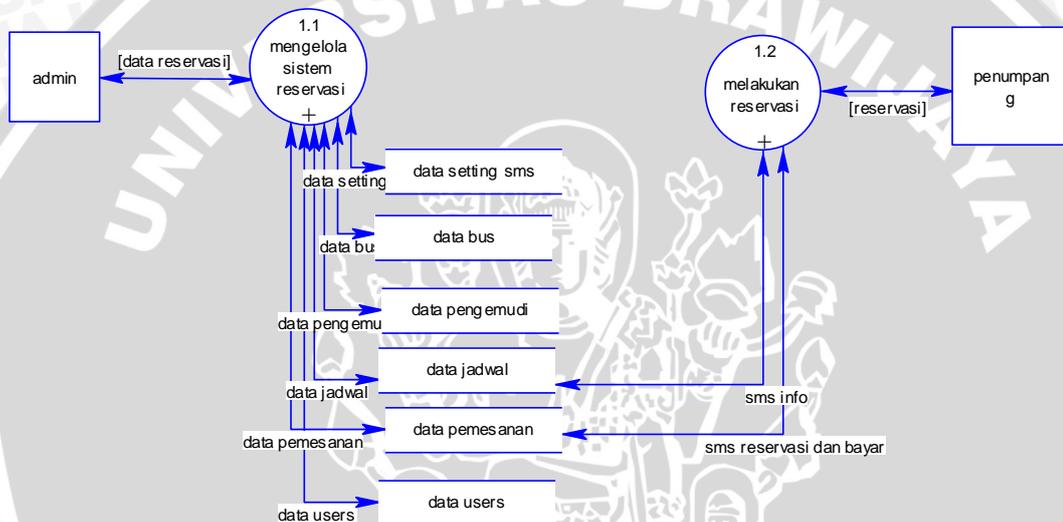
Sumber : [Perancangan]

## 4.2.2 Pemodelan Dengan DFD



**Gambar 4.19 DFD Level 0 Proses Reservasi Tiket**  
Sumber : [Perancangan]

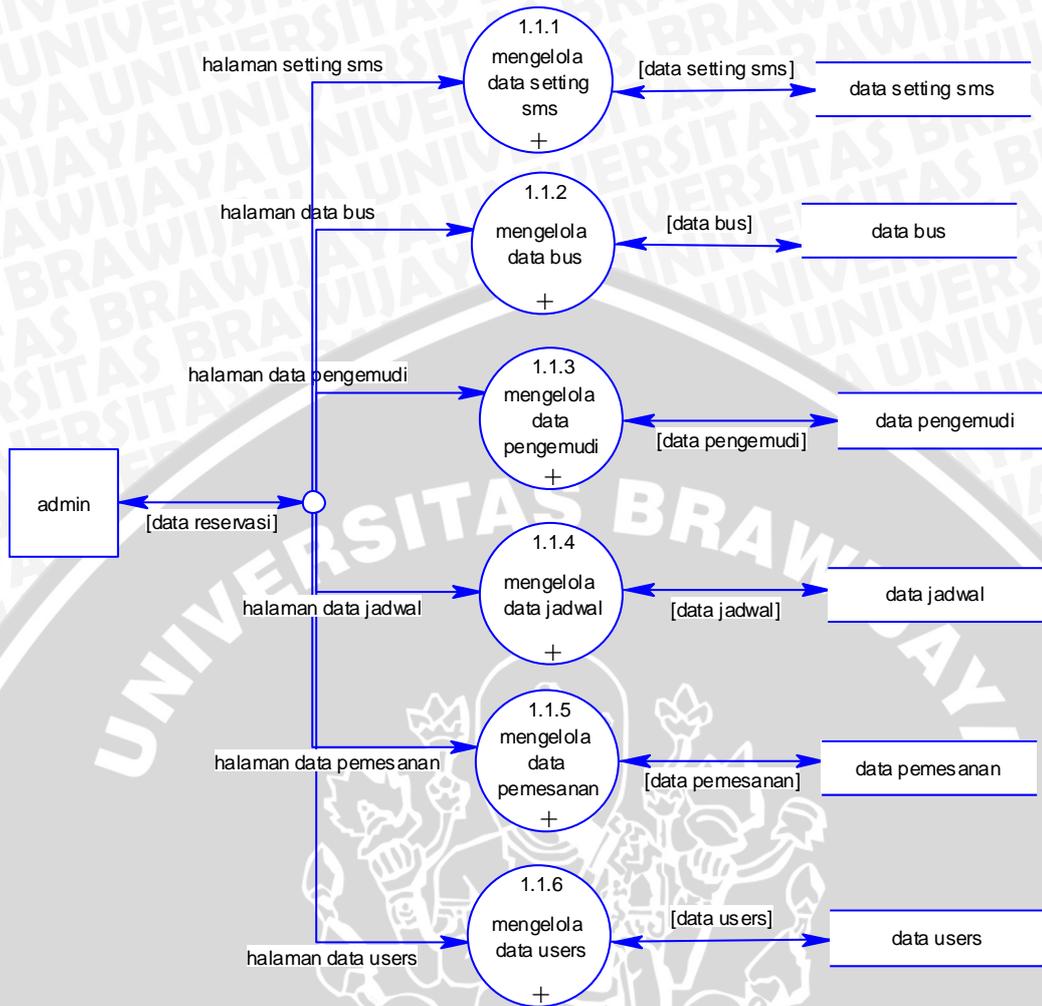
Gambar 4.19 menggambarkan sistem reservasi tiket bus di Terminal Arjosari Malang secara garis besar pada level 0.



**Gambar 4.20 DFD Level 1 Proses Reservasi Tiket**  
Sumber : [Perancangan]

Gambar 4.20 mempunyai keterangan sebagai berikut :

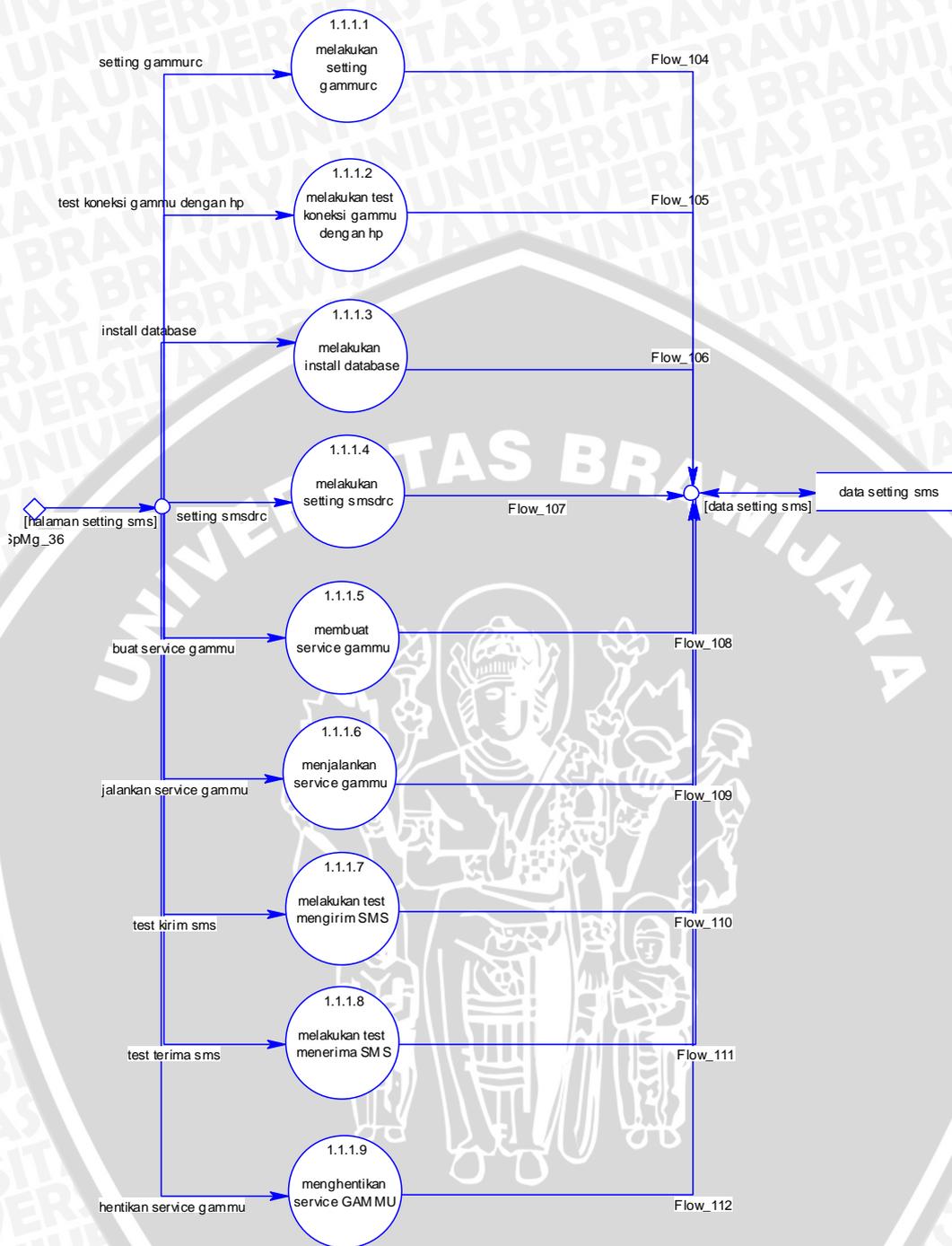
1. Proses 1.1 admin melakukan proses mengelola sistem reservasi.
2. Proses 1.2 penumpang melakukan proses reservasi.
3. Didalam sistem reservasi terdapat beberapa datastore yang masing-masing berfungsi sebagai tempat penyimpanan data yaitu data setting sms, data bus, data pengemudi, data jadwal, data pemesanan dan data users.



**Gambar 4.21 DFD Level 1 (lanjutan) Proses Mengelola Sistem Reservasi**  
**Sumber : [Perancangan]**

Gambar 4.21 mempunyai keterangan sebagai berikut :

Admin melakukan beberapa proses didalam mengelola sistem reservasi yaitu mengelola data setting sms, mengelola data bus, mengelola data pengemudi, mengelola data jadwal, mengelola data pemesanan dan mengelola data users.



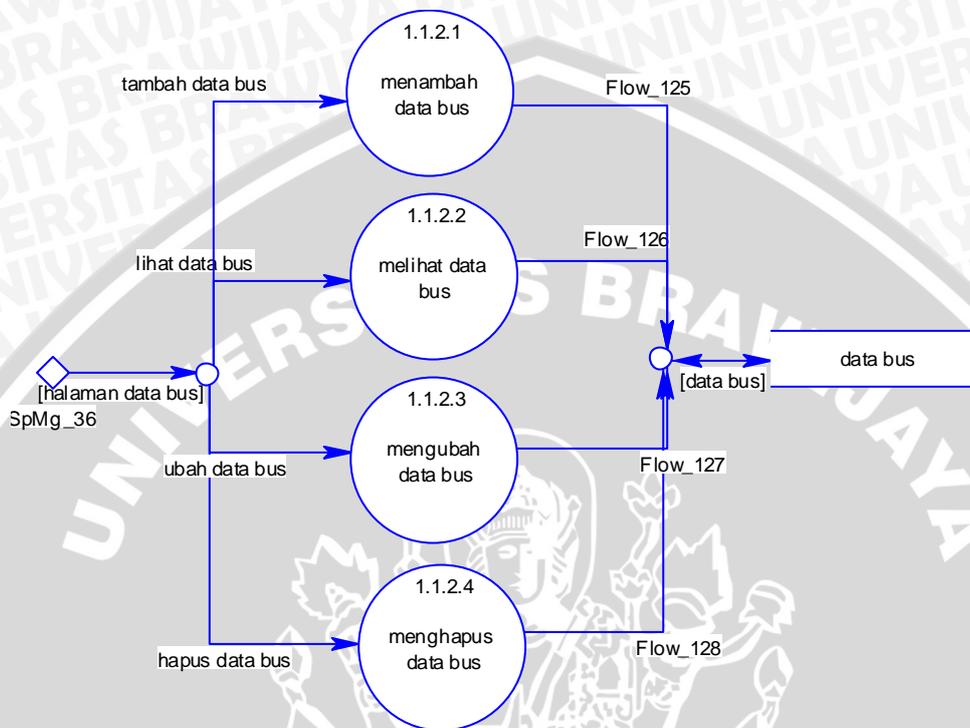
**Gambar 4.22 DFD Level 1 (lanjutan) Mengelola Data Setting SMS**  
 Sumber : [Perancangan]

Gambar 4.22 mempunyai keterangan sebagai berikut :

1. Didalam proses mengelola data setting sms ada beberapa proses lagi yang dapat dilakukan oleh admin yaitu melakukan setting gammurc, melakukan test koneksi gammu dengan hp, melakukan install database, melakukan setting smsdrc, membuat service gammu, menjalankan service gammu,

melakukan test mengirim sms, melakukan test menerima sms dan menghentikan service gammu.

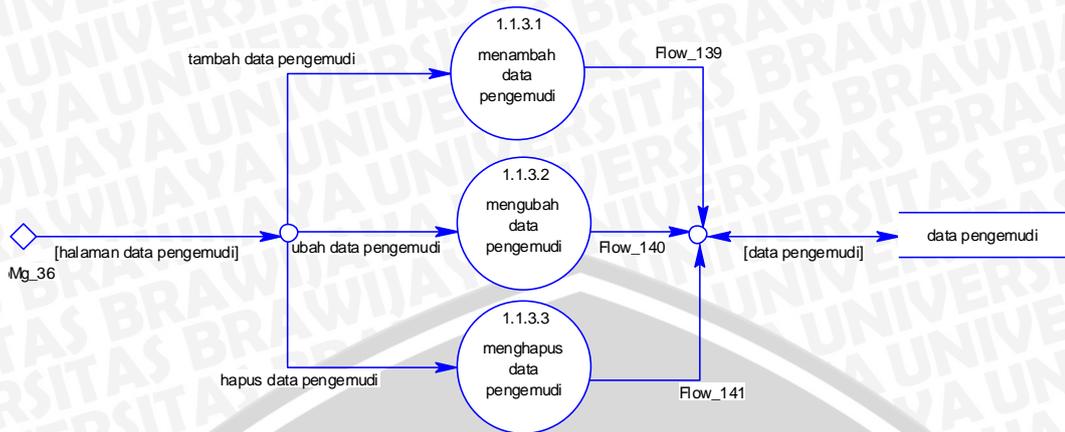
2. Data setting sms terdiri dari beberapa data yaitu daemons, gammu, inbox, outbox, outbox\_multipart, pbk, pbk\_groups, phones, sentitems.



**Gambar 4.23 DFD Level 1 (lanjutan) Mengelola Data Bus**  
 Sumber : [Perancangan]

Gambar 4.23 mempunyai keterangan sebagai berikut :

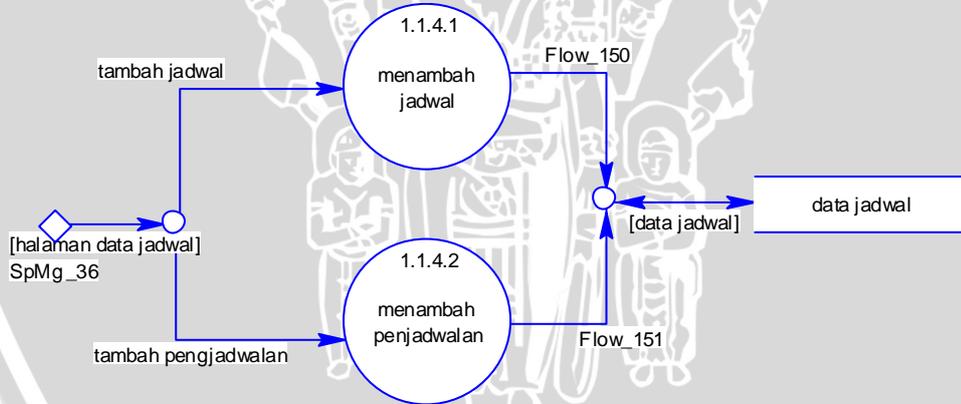
1. Didalam proses mengelola data bus ada beberapa proses lagi yang dapat dilakukan oleh admin yaitu menambah data bus, melihat data bus, mengubah data bus dan menghapus data bus.
2. Data bus terdiri dari beberapa data yaitu kode\_bus, no\_jadwal, id\_pengemudi, nama\_bus, gambar\_bus, harga dan keterangan.



**Gambar 4.24 DFD Level 1 (lanjutan) Mengelola Data Pengemudi**  
 Sumber : [Perancangan]

Gambar 4.24 mempunyai keterangan sebagai berikut :

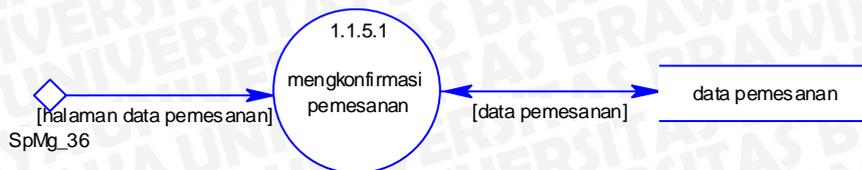
1. Didalam proses mengelola data pengemudi ada beberapa proses lagi yang dapat dilakukan oleh admin yaitu menambah data pengemudi, mengubah data pengemudi dan menghapus data pengemudi.
2. Data pengemudi terdiri dari beberapa data yaitu id\_pengemudi, kode\_bus, nama pengemudi, alamat\_pengemudi dan no\_telp.



**Gambar 4.25 DFD Level 1 (lanjutan) Mengelola Data Jadwal**  
 Sumber : [Perancangan]

Gambar 4.25 mempunyai keterangan sebagai berikut :

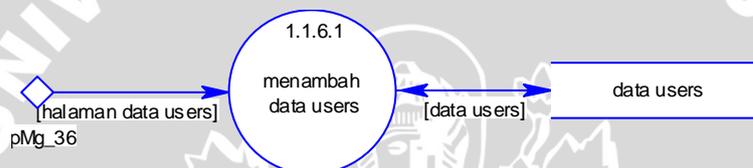
1. Didalam proses mengelola data jadwal ada beberapa proses lagi yang dapat dilakukan oleh admin yaitu menambah jadwal dan menambah penjadwalan.
2. Data jadwal terdiri dari beberapa data yaitu no\_jadwal, tujuan, tgl\_berangkat, jam\_berangkat, jam\_tiba.



**Gambar 4.26 DFD Level 1 (lanjutan) Mengelola Data Pemesanan**  
**Sumber : [Perancangan]**

Gambar 4.26 mempunyai keterangan sebagai berikut :

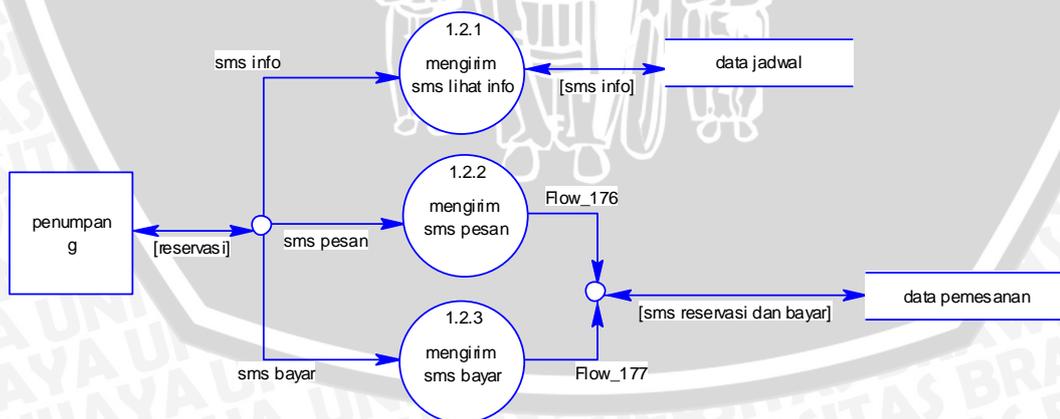
1. Didalam proses mengelola data pemesanan admin melakukan proses mengkonfirmasi pemesanan.
2. Data pemesanan terdiri dari beberapa data yaitu nomor\_pesan, tanggal\_pesan, kode\_bus, nama, alamat, email, telepon, jumlah\_tiket, via, status\_konfirmasi, status\_bayar, status\_sms dan status\_smsbayar.



**Gambar 4.27 DFD Level 1 (lanjutan) Mengelola Data Users**  
**Sumber : [Perancangan]**

Gambar 4.27 mempunyai keterangan sebagai berikut :

1. Didalam proses mengelola data users admin melakukan proses menambah data users.
2. Data users terdiri dari beberapa data yaitu usernames dan password.



**Gambar 4.28 DFD Level 1 (lanjutan) Proses Melakukan Reservasi**  
**Sumber : [Perancangan]**

Gambar 4.28 mempunyai keterangan sebagai berikut :

1. penumpang melakukan beberapa proses didalam melakukan reservasi yaitu mengirim sms lihat info, mengirim sms pesan dan mengirim sms bayar.
2. format sms lihat info : info (spasi) nama bus (spasi) kota tujuan (spasi) tanggal berangkat.
3. format sms pesan : pesan (spasi) nama bus (spasi) kota tujuan (spasi) tanggal berangkat (spasi) jumlah tiket.
4. format sms bayar : bayar (spasi) no order (spasi) no bukti bank (spasi) tanggal kirim.

### 4.3 PERANCANGAN ANTARMUKA

Pada bagian ini akan dijelaskan tentang perancangan antarmuka perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang. Antarmuka perangkat lunak ini akan digunakan oleh pengguna untuk berinteraksi dengan sistem perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang.

#### 4.3.1 Perancangan Alur Kerja Penumpang

Perancangan alur kerja penumpang SMS Gateway pada Sistem Reservasi Tiket terdiri dari perancangan alur kerja penumpang melihat info via SMS, melihat info via *Website*, memesan tiket via SMS, memesan tiket via *Website* dan mengkonfirmasi pembayaran.

##### a. Perancangan Alur kerja Melihat Info via SMS

Perancangan alur kerja Melihat Info via SMS mengacu pada spesifikasi kebutuhan SRS\_001\_01 seperti pada Gambar 4.29.

#### NAMA ALUR KERJA Melihat Info via SMS

##### DESKRIPSI

Masukan :

1. SMSs

Proses :

1. Penumpang mengirim SMS "info (spasi) nama bus (spasi) kota tujuan (spasi) tanggal berangkat"

Keluaran :

1. Penumpang menerima sms balasan berupa jadwal info bus yang dicari.

**Gambar 4.29 Perancangan Alur kerja Melihat Info Via SMS**  
Sumber: [Perancangan]

**b. Perancangan Alur kerja Melihat Info via Website**

Perancangan alur kerja Melihat Info via *Website* mengacu pada spesifikasi kebutuhan SRS\_001\_02 seperti pada Gambar 4.30.

**NAMA ALUR KERJA** Melihat Info via *Website*

**DESKRIPSI**

Masukan :

1. Buka *Website*
2. Tanggal
3. Tujuan

Proses :

1. Penumpang membuka *Website*.
2. Penumpang memilih tanggal pada kolom “Masukkan Tanggal”.
3. Penumpang memilih tujuan pada kolom “Pilih Tujuan”.
4. Penumpang menekan tombol “Cari”

Keluaran :

1. Sistem akan menampilkan data jadwal bus sesuai yang dicari penumpang.

**Gambar 4.30 Perancangan Alur kerja Melihat Info Via Website**  
Sumber: [Perancangan]

**c. Perancangan Alur kerja Memesan Tiket via SMS**

Perancangan alur kerja Memesan Tiket via SMS mengacu pada spesifikasi kebutuhan SRS\_001\_03 seperti pada Gambar 4.31.

**NAMA ALUR KERJA** Memesan Tiket via SMS

**DESKRIPSI**

Masukan :

1. SMS

Proses :

1. Penumpang mengirim SMS “pesan (spasi) nama bus (spasi) kota tujuan (spasi) tanggal berangkat (spasi) jumlah tiket”.

Keluaran :

1. Penumpang menerima sms balasan berupa konfirmasi pemesanan tiket.

**Gambar 4.31 Perancangan Alur kerja Memesan Tiket via SMS**  
Sumber: [Perancangan]

**d. Perancangan Alur kerja Memesan Tiket via Website**

Perancangan alur kerja Memesan Tiket via *Website* mengacu pada spesifikasi kebutuhan SRS\_001\_04 seperti pada Gambar 4.32.

**NAMA ALUR KERJA** Memesan Tiket via *Website***DESKRIPSI**

Masukan :

1. Buka *Website*
2. Pilih menu pemesanan via online
3. Tanggal
4. Tujuan
5. Nama
6. Alamat
7. Email
8. No Telp/hp
9. Jumlah tiket

Proses :

1. Penumpang membuka *Website*.
2. Penumpang memilih pemesanan via online.
3. Penumpang memilih tanggal pada kolom "Masukkan Tanggal"
4. Penumpang memilih tujuan pada kolom "Tujuan".
5. Penumpang menekan tombol "Cari"
6. Penumpang menekan tombol "Pesan"
7. Penumpang mengisi nama pada kolom "Nama"
8. Penumpang mengisi alamat pada kolom "Alamat".
9. Penumpang mengisi email pada kolom "Email".
10. Penumpang mengisi nomor telp/hp pada kolom "Telp/hp".
11. Penumpang mengisi jumlah tiket pada kolom "Jumlah Tiket".
12. Penumpang menekan tombol "Pesan".

Keluaran :

1. Sistem akan menampilkan pesan konfirmasi pembayaran dan silahkan melakukan pembayaran.

**Gambar 4.32** Perancangan Alur kerja Memesan Tiket via *Website*

Sumber: [Perancangan]

**e. Perancangan Alur kerja Mengkonfirmasi Pembayaran via SMS**

Perancangan alur kerja Mengkonfirmasi Pembayaran via SMS mengacu pada spesifikasi kebutuhan SRS\_001\_05 seperti pada Gambar 4.33.

**NAMA ALUR KERJA** Mengkonfirmasi Pembayaran via SMS**DESKRIPSI**

Masukan :

1. SMS

Proses :

1. Penumpang mengirim SMS "bayar (spasi) no.order (spasi) no. bukti bank (spasi) tanggal kirim".

Keluaran :

1. Penumpang menerima sms balasan berupa konfirmasi pembayaran.

**Gambar 4.33** Perancangan Alur kerja Mengkonfirmasi Pembayaran via SMS

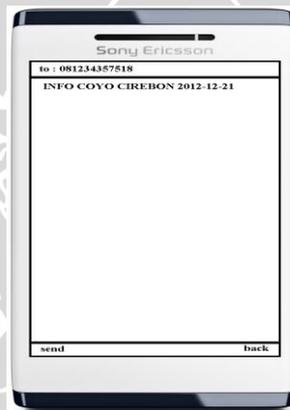
Sumber: [Perancangan]

### 4.3.2 Perancangan Antarmuka Penumpang

Antarmuka pengguna untuk aplikasi penumpang berupa sebuah format SMS sistem reservasi berbasis SMS Gateway. Antarmuka aplikasi penumpang terdiri dari SMS Melihat Info, SMS Info Bus, SMS Pemesanan Tiket, SMS Konfirmasi Pemesanan Tiket Berhasil, SMS Konfirmasi Bayar.

#### a. SMS Melihat Info

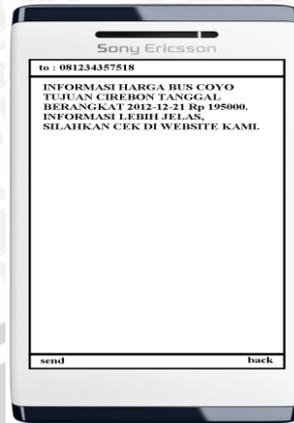
SMS Melihat Info merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. Format SMS ini berfungsi mempermudah penumpang untuk mengetahui info bus melalui SMS. Perancangan antarmuka ini mengacu pada spesifikasi kebutuhan SRS\_001\_01. Gambar 4.34 akan menunjukkan perancangan tampilan antarmuka dari SMS Melihat Info.



Gambar 4.34 Tampilan Antarmuka SMS Melihat Info  
Sumber : [Perancangan]

#### b. SMS Info Bus

SMS Info Bus merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. SMS ini merupakan balasan dari sistem tentang informasi bus yang dicari oleh penumpang. Gambar 4.35 akan menunjukkan perancangan tampilan antarmuka dari SMS Info Bus.

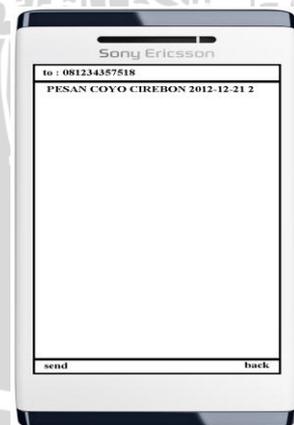


**Gambar 4.35 Tampilan Antarmuka SMS Informasi Bus**  
Sumber : [Perancangan]

Gambar 4.35 menampilkan format SMS yang diterima penumpang, apabila bus yang dicari tersedia.

### **c. SMS Pemesanan Tiket**

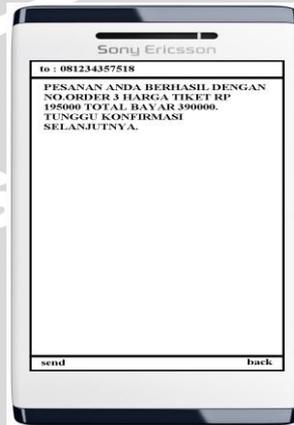
SMS Pemesanan Tiket merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. Format SMS ini berfungsi mempermudah penumpang untuk memesan tiket melalui SMS. Perancangan antarmuka ini mengacu pada spesifikasi kebutuhan SRS\_001\_02. Gambar 4.36 akan menunjukkan perancangan tampilan antarmuka dari SMS Memesan Tiket.



**Gambar 4.36 Tampilan Antarmuka SMS Memesan Tiket**  
Sumber : [Perancangan]

**d. SMS Konfirmasi Pemesanan Tiket Berhasil**

SMS Konfirmasi Pemesanan Tiket Berhasil merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. SMS ini merupakan balasan dari sistem tentang konfirmasi bus yang dipesan oleh penumpang. Gambar 4.37 akan menunjukkan perancangan tampilan antarmuka dari SMS Konfirmasi Pemesanan Tiket Berhasil.

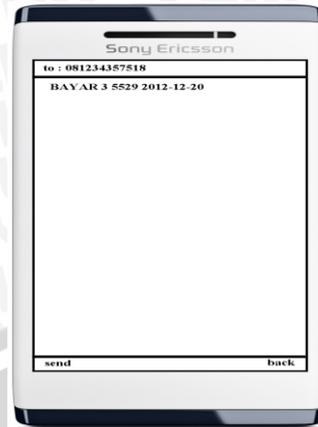


**Gambar 4.37 Tampilan Antarmuka SMS Konfirmasi Pemesanan Tiket Berhasil**  
Sumber : [Perancangan]

Gambar 4.37 menampilkan format SMS yang diterima penumpang, apabila tiket yang dipesan tersedia.

**e. SMS Konfirmasi Bayar**

SMS Konfirmasi Bayar merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. Format SMS ini berfungsi mempermudah penumpang untuk mengkonfirmasi pembayaran pemesanan tiket melalui SMS. Perancangan antarmuka ini mengacu pada spesifikasi kebutuhan SRS\_001\_04. Gambar 4.38 akan menunjukkan perancangan tampilan antarmuka dari SMS Konfirmasi Bayar.



**Gambar 4.38 Tampilan Antarmuka SMS Konfirmasi Bayar**  
Sumber : [Perancangan]

### 4.3.3 Perancangan Alur Kerja Admin

Perancangan alur kerja admin SMS Gateway pada Sistem Reservasi Tiket terdiri dari perancangan alur kerja admin login, menjalankan service SMS GATEWAY, mengelola data pengemudi, mengelola data nus, mengelola data jadwal, mengelola data pemesanan dan mengelola data users.

#### a. Perancangan Alur kerja Admin Login

Perancangan alur kerja Login mengacu pada spesifikasi kebutuhan SRS\_002\_01 seperti pada Gambar 4.39.

##### **NAMA ALUR KERJA** Login

##### **DESKRIPSI**

Masukan : Username dan Password

Proses :

1. Admin mengisi username pada kolom "Username".
2. Admin mengisi password pada kolom "Password".
3. Admin menekan Tombol "Login".
4. Sistem akan melakukan seleksi kondisi jika masukan bernilai kosong maka akan ditampilkan pesan "Peringatan", sedangkan jika masukan bernilai benar, maka akan ditampilkan halaman Beranda Admin.

Keluaran :

1. Halaman Beranda aplikasi Admin tampil di layar.

**Gambar 4.39 Perancangan Alur kerja Login**  
Sumber: [Perancangan]

#### b. Perancangan Alur kerja Admin Menjalankan Service SMS Gateway

Perancangan alur kerja Menjalankan Service SMS GATEWAY mengacu pada spesifikasi kebutuhan SRS\_002\_02 seperti pada Gambar 4.40.

**NAMA ALUR KERJA** Menjalankan Service SMS GATEWAY

**DESKRIPSI**

Masukan :

1. Pilih menu Run SMS

Proses :

1. Admin memilih menu “Run SMS”.
2. Admin menekan tombol “Jalankan Service”.
3. Admin memilih menu service yang akan dijalankan

Keluaran :

1. Service yang dijalankan tampil dilayar.

**Gambar 4.40** Perancangan Alur kerja Menjalankan Service SMS GATEWAY

Sumber: [Perancangan]

**c. Perancangan Alur kerja Admin Mengelola Data Pengemudi**

Perancangan alur kerja Mengelola Data Pengemudi bertujuan untuk melakukan manipulasi data pengemudi yang disimpan di *database*. Perancangan alur kerja Mengelola Data Pengemudi mengacu pada spesifikasi kebutuhan SRS\_002\_03. Pengguna yang terdaftar sebagai admin dapat melihat, menambah, mengubah, dan menghapus data pengemudi yang digunakan aplikasi sistem reservasi tiket ini. Perancangan alur kerja Lihat Pengemudi seperti pada Gambar 4.41.

**NAMA ALUR KERJA** Lihat Pengemudi

**DESKRIPSI**

Masukan : Pilih menu Pengemudi

Proses :

1. Admin memilih menu “Pengemudi”.

Keluaran :

1. Semua data pengemudi ditampilkan pada halaman Pengemudi.
2. Halaman Admin tampil di layar.

**Gambar 4.41** Perancangan Alur kerja Lihat Pengemudi

Sumber: [Perancangan]

Perancangan alur kerja Tambah Pengemudi seperti pada Gambar 4.42.

**NAMA ALUR KERJA** Tambah Pengemudi  
**DESKRIPSI**

Masukan :

1. Pilih menu Halaman Pengemudi
2. Id Pengemudi
3. Nama Pengemudi
4. Alamat Pengemudi
5. No.telp Pengemudi

Proses :

1. Admin memilih menu Pengemudi.
2. Admin mengisi Id pengemudi pada kolom “Id Pengemudi”.
3. Admin mengisi Nama pengemudi pada kolom “Nama Pengemudi”.
4. Admin mengisi Alamat pengemudi pada kolom “Alamat”.
5. Admin mengisi No.telp pengemudi pada kolom “No.Telp”.
6. Admin menekan tombol “Simpan”
7. Sistem akan melakukan seleksi kondisi jika masukan bernilai kosong maka akan ditampilkan pesan “Peringatan”, sedangkan jika masukan bernilai benar, maka akan ditampilkan data pengemudi di Halaman Pengemudi.

Keluaran :

1. Halaman Pengemudi tampil di layar.

**Gambar 4.42 Perancangan Alur kerja Tambah Pengemudi**

Sumber: [Perancangan]

Perancangan alur kerja Ubah Pengemudi seperti pada Gambar 4.43.

**NAMA ALUR KERJA** Ubah Pengemudi  
**DESKRIPSI**

Masukan : Pilih menu Halaman Pengemudi dan pilih tombol “Ubah” pada Pengemudi yang ingin diubah.

Proses :

1. Admin memilih menu Pengemudi.
2. Admin memilih data pengemudi yang akan diubah.
3. Detail pengemudi akan ditampilkan pada layar.
4. Admin mengisi Id pengemudi pada kolom “Id Pengemudi”.
5. Admin mengisi Nama pengemudi pada kolom “Nama Pengemudi”.
6. Admin mengisi Alamat pengemudi pada kolom “Alamat”.
7. Admin mengisi No.telp pengemudi pada kolom “No.Telp”.
8. Admin menekan tombol “Simpan”
9. Sistem akan melakukan seleksi kondisi jika masukan bernilai kosong maka akan ditampilkan pesan “Peringatan”, sedangkan jika masukan bernilai benar, maka akan ditampilkan data pengemudi yang telah ditambahkan di Halaman Pengemudi.

Keluaran :

1. Halaman Pengemudi tampil di layar.

**Gambar 4.43 Perancangan Alur kerja Ubah Pengemudi**

Sumber: [Perancangan]

Perancangan alur kerja Hapus Admin seperti pada Gambar 4.44.

**NAMA ALUR KERJA** Hapus Pengemudi  
**DESKRIPSI**

Masukan : Pilih menu Pengemudi dan pilih tombol “Hapus” pada pengemudi yang akan di hapus.

Proses :

1. Admin memilih menu Pengemudi.
2. Admin memilih data pengemudi yang akan dihapus dan menekan tombol “Hapus”.

Keluaran :

1. Halaman Pengemudi tampil di layar.

**Gambar 4.44 Perancangan Alur kerja Hapus Pengemudi**  
 Sumber: [Perancangan]

**d. Perancangan Alur kerja Admin Mengelola Data Bus**

Perancangan alur kerja Mengelola Data Bus bertujuan untuk melakukan manipulasi data bus yang disimpan di *database*. Perancangan alur kerja Mengelola Data Bus mengacu pada spesifikasi kebutuhan SRS\_002\_04. Pengguna yang terdaftar sebagai admin dapat melihat, menambah, dan menghapus data bus pada perangkat lunak sms gateway ini. Perancangan algoritma Lihat Bus seperti pada Gambar 4.45.

**NAMA ALUR KERJA** Lihat Bus  
**DESKRIPSI**

Masukan : Pilih menu Bus

Proses :

1. Admin memilih menu “Bus”.

Keluaran :

1. Semua data bus ditampilkan pada halaman Bus.
2. Halaman Bus tampil di layar.

**Gambar 4.45 Perancangan Alur kerja Lihat Bus**  
 Sumber: [Perancangan]

Perancangan alur kerja Tambah Bus seperti pada Gambar 4.46.

**NAMA ALUR KERJA** Tambah Bus  
**DESKRIPSI**

Masukan :

1. Pilih menu Halaman Bus
2. Kode Bus
3. Nama Bus
4. Pengemudi
5. Harga Bus
6. Gambar bus
7. Keterangan

Proses :

1. Admin memilih menu Bus.
2. Admin mengisi Kode Bus pada kolom “Kode Bus”.
3. Admin mengisi Nama Bus pada kolom “Nama Bus”.
4. Admin memilih Pengemudi pada kolom “Pengemudi”.
5. Admin mengisi Harga Bus pada kolom “Harga Bus”.
6. Admin memilih Gambar Bus pada kolom “Gambar Bus”.
7. Admin mengisi Keterangan pada kolom “Keterangan”.
8. Admin menekan tombol Simpan.
9. Sistem akan melakukan seleksi kondisi jika masukan bernilai kosong maka akan ditampilkan pesan “Peringatan”, sedangkan jika masukan bernilai benar, maka akan ditampilkan data bus yang telah ditambahkan di Halaman Bus.

Keluaran :

1. Halaman Bus tampil di layar.

**Gambar 4.46 Perancangan Alur kerja Tambah Bus**  
Sumber: [Perancangan]

Perancangan alur kerja Ubah Bus seperti gambar 4.47.

**NAMA ALUR KERJA** Ubah Bus

**DESKRIPSI**

Masukan : Pilih menu Halaman Bus dan pilih tombol “Ubah” pada Bus yang ingin diubah.

Proses :

1. Admin memilih menu Bus.
2. Admin memilih data bus yang akan diubah.
3. Detail bus akan ditampilkan pada layar.
4. Admin mengisi Kode Bus pada kolom “Kode Bus”.
5. Admin mengisi Nama Bus pada kolom “Nama Bus”.
6. Admin memilih Pengemudi pada kolom “Pengemudi”.
7. Admin mengisi Harga Bus pada kolom “Harga Bus”.
8. Admin memilih Gambar Bus pada kolom “Gambar Bus”.
9. Admin mengisi Keterangan pada kolom “Keterangan”.
10. Admin menekan tombol “Simpan”
11. Sistem akan melakukan seleksi kondisi jika masukan bernilai kosong maka akan ditampilkan pesan “Peringatan”, sedangkan jika masukan bernilai benar, maka akan ditampilkan data bus yang telah ditambahkan di Halaman Bus.

Keluaran :

1. Halaman Bus tampil di layar.

**Gambar 4.47 Perancangan Alur kerja Ubah Bus**  
Sumber: [Perancangan]

Perancangan alur kerja Hapus Bus seperti pada Gambar 4.48.

**NAMA ALUR KERJA** Hapus Bus

**DESKRIPSI**

Masukan : Pilih menu Bus dan pilih tombol “Hapus” pada bus yang akan di hapus.

Proses :

1. Admin memilih menu Bus.

2. Admin memilih data bus yang akan dihapus dan menekan tombol “Hapus”.
- Keluaran :
1. Halaman Bus tampil di layar.

**Gambar 4.48 Perancangan Alur kerja Hapus Bus**  
Sumber: [Perancangan]

#### e. Perancangan Alur kerja Admin Mengelola Data Jadwal Bus

Perancangan alur kerja Mengelola Data Jadwal Bus bertujuan untuk melakukan manipulasi data jadwal bus yang disimpan di *database*. Perancangan alur kerja Mengelola Data Jadwal Bus mengacu pada spesifikasi kebutuhan SRS\_002\_05. Pengguna yang terdaftar sebagai admin dapat melihat, menambah, mengubah, dan menghapus data jadwal bus yang terdapat pada Terminal Arjosari Malang. Perancangan alur kerja Lihat Jadwal Bus seperti pada Gambar 4.49.

##### **NAMA ALUR KERJA** Lihat Jadwal Bus

##### **DESKRIPSI**

Masukan : Pilih menu Jadwal

Proses :

1. Admin memilih menu “Jadwal”.

Keluaran :

1. Semua data jadwal bus pada hari itu ditampilkan pada halaman Jadwal.
2. Halaman Jadwal Bus tampil di layar.

**Gambar 4.49 Perancangan Alur kerja Lihat Jadwal Bus**  
Sumber: [Perancangan]

Perancangan alur kerja Tambah Jadwal Bus seperti pada Gambar 4.50.

##### **NAMA ALUR KERJA** Tambah Jadwal Bus

##### **DESKRIPSI**

Masukan :

1. Pilih menu Jadwal
2. Tujuan
3. Tanggal Berangkat
4. Jam Berangkat
5. Jam Tiba
6. Kode Bus
7. Id Jadwal

Proses :

1. Admin memilih menu Jadwal.
2. Admin menekan tombol “Tambah Jadwal”.
3. Admin memilih Tujuan pada kolom “Tujuan”.
4. Admin mengisi Tanggal Berangkat pada kolom “Tanggal Berangkat”
5. Admin mengisi Jam Berangkat pada kolom “Jam Berangkat”.
6. Admin mengisi Jam Tiba pada kolom “Jam Tiba”.
7. Admin menekan Tombol “Simpan”
8. Admin menekan Tombol “Tambah Penjadwalan”.

9. Admin memilih Kode Bus pada kolom “Kode Bus”.
10. Admin memilih Id Jadwal pada kolom “Id Jadwal”.
11. Admin menekan tombol “Simpan”
12. Sistem akan melakukan seleksi kondisi jika masukan bernilai kosong maka akan ditampilkan pesan “Peringatan”, sedangkan jika masukan bernilai benar, maka akan ditampilkan data jadwal yang telah ditambahkan di Halaman Jadwal.

Keluaran :

1. Halaman Jadwal Bus tampil di layar.

**Gambar 4.50 Perancangan Alur kerja Tambah Jadwal Bus**  
Sumber: [Perancangan]

#### **f. Perancangan Alur kerja Admin Mengelola Data Pemesanan**

Perancangan alur kerja Mengelola Data Pemesanan bertujuan untuk mempermudah pengguna yang terdaftar sebagai *admin* untuk melihat Pemesanan yang masuk sistem dari aplikasi ini. Perancangan alur kerja Mengelola Data Pemesanan mengacu pada spesifikasi kebutuhan SRS\_002\_06. Perancangan alur kerja Lihat Halaman Pemesanan seperti pada Gambar 4.51.

**NAMA ALUR KERJA** Lihat Halaman Pemesanan

#### **DESKRIPSI**

Masukan : Pilih menu Pemesanan

Proses :

1. Admin memilih menu “Pemesanan”
2. Sistem akan menampilkan Pemesanan secara berurutan, sesuai dengan dataPemesanan yang terdapat pada basis data.

Keluaran :

1. Halaman Pemesanan aplikasi admin tampil di layar.

**Gambar 4.51 Perancangan Alur kerja Lihat Pemesanan**  
Sumber: [Perancangan]

Perancangan alur kerja Konfirmasi Pemesanan seperti pada Gambar 4.52.

**NAMA ALUR KERJA** Lihat Halaman Pemesanan

**DESKRIPSI**

Masukan :

1. Pilih menu Pemesanan
2. Pilih menu Pemesanan Belum Dikonfirmasi
3. Pilih menu Pemesanan Sudah Dikonfirmasi
4. Pilih menu Pemesanan Ditolak

Proses :

1. Admin memilih menu “Pemesanan”.
2. Admin memilih menu “Pemesanan Belum Dikonfirmasi”.
3. Admin memilih Status Konfirmasi pada kolom “Status Konfirmasi”.
4. Admin menekan tombol “update”.
5. Admin memilih menu “Pemesanan Sudah Dikonfirmasi”.
6. Admin memilih Status Pembayaran pada kolom “Status Pembayaran”.
7. Admin menekan tombol “Update”.
8. Admin memilih Menu “Pemesanan Ditolak”.
9. Sistem akan menampilkan data Pemesanan Belum Dikonfirmasi, Pemesanan Sudah Dikonfirmasi dan Pemesanan Ditolak pada Halaman Pemesanan

Keluaran :

1. Halaman Pemesanan aplikasi admin tampil di layar.

**Gambar 4.52 Perancangan Alur kerja Konfirmasi Pemesanan**  
Sumber: [Perancangan]

**g. Perancangan Alur kerja Admin Halaman Users**

Perancangan alur kerja Halaman Users bertujuan untuk menambah user admin aplikasi ini. Perancangan alur kerja Halaman Users pada aplikasi admin mengacu pada spesifikasi kebutuhan SRS\_002\_07. Perancangan alur kerja Lihat Halaman Users seperti pada Gambar 4.53.

**NAMA ALUR KERJA** Lihat Halaman Users

**DESKRIPSI**

Masukan : Pilih menu Users

Proses :

1. Admin memilih menu “Users”
2. Sistem akan menampilkan data User pada Halaman Users.

Keluaran :

1. Halaman Users aplikasi admin tampil di layar.

**Gambar 4.53 Perancangan Alur kerja Lihat Users**  
Sumber: [Perancangan]

Perancangan alur kerja Tambah User seperti pada Gambar 4.54.

#### **NAMA ALUR KERJA** Tambah User

##### **DESKRIPSI**

Masukan :

1. Pilih menu User
2. Username
3. Password
4. Retry Password

Proses :

1. Admin memilih menu “Users”.
2. Admin menekan tombol “Tambah User”.
3. Admin mengisi Username pada kolom “Username”.
4. Admin mengisi Password pada kolom “Password”.
5. Admin mengisi Password pada kolom “Retry Password”.

Keluaran :

1. Sistem akan melakukan seleksi kondisi jika masukan bernilai kosong maka akan ditampilkan pesan “Peringatan”, sedangkan jika masukan bernilai benar, maka akan ditampilkan data user yang telah ditambahkan di Halaman Users.

**Gambar 4.54 Perancangan Alur kerja Tambah Users**

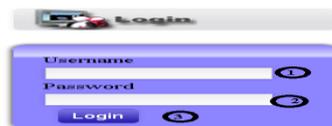
Sumber: [Perancangan]

### **4.3.4 Perancangan Antarmuka Admin**

Antarmuka pengguna untuk aplikasi admin berupa sebuah halaman *website*. Antarmuka aplikasi admin terdiri dari halaman *Login*, halaman *Home*, halaman *Run SMS*, keluar, halaman Pengemudi, halaman Bus, halaman Jadwal, halaman Pemesanan dan halaman *Users*.

#### **a. Halaman Login**

Halaman login merupakan salah satu antarmuka pengguna untuk aplikasi admin. Halaman *login* berfungsi mempermudah pengguna yang terdaftar sebagai admin untuk mengelola aplikasi ini. Gambar 4.55 akan menunjukkan perancangan tampilan antarmuka dari halaman *Login*.



**Gambar 4.55 Tampilan Antarmuka Halaman Login**

Sumber : [perancangan]

Gambar 4.55 memiliki keterangan sebagai berikut :

1. Kolom *Username* untuk mengisi nama pengguna.
2. Kolom *Password* untuk mengisi kata sandi.
3. Tombol *Login* untuk menjalankan proses *login*.

**b. Halaman Home**

Halaman *home* berfungsi bagi admin untuk mengelola *content website* dan mengelola akun admin.



**Gambar 4.56 Tampilan Antarmuka Halaman Home**  
Sumber : [Perancangan]

Gambar 4.56 memiliki keterangan sebagai berikut :

1. Tombol *Home* untuk menuju ke halaman utama.
2. Tombol *Run SMS* untuk menjalankan SMS GATEWAY.
3. Tombol *Logout* untuk keluar dari halaman admin.
4. Tombol Edit untuk mengubah *password* akun.
5. Tombol Pengemudi untuk mengelola data pengemudi.
6. Tombol Bus untuk mengelola data bus.
7. Tombol Jadwal untuk mengelola jadwal bus.
8. Tombol Pemesanan untuk mengelola data pemesanan tiket.
9. Tombol Users untuk mengelola data admin.
10. Tampilan Hari dan tanggal untuk menunjukkan hari dan tanggal saat itu.

c. **Halaman Run SMS**

Halaman *run* SMS berfungsi bagi admin untuk menjalankan program SMS GATEWAY.



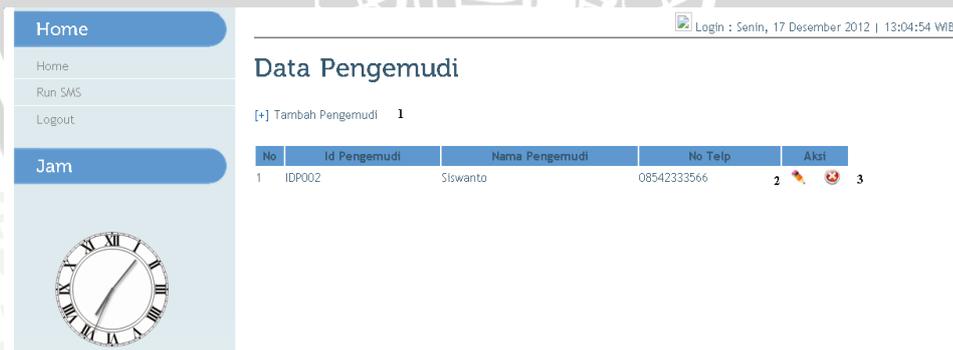
**Gambar 4.57 Tampilan Antarmuka Halaman Run SMS**  
Sumber : [Perancangan]

Gambar 4.57 memiliki keterangan sebagai berikut :

1. Menu Jalankan Service untuk memulai Service SMS Gateway.

d. **Halaman Pengemudi**

Halaman pengemudi berfungsi bagi admin untuk menambah, mengubah dan menghapus data pengemudi.



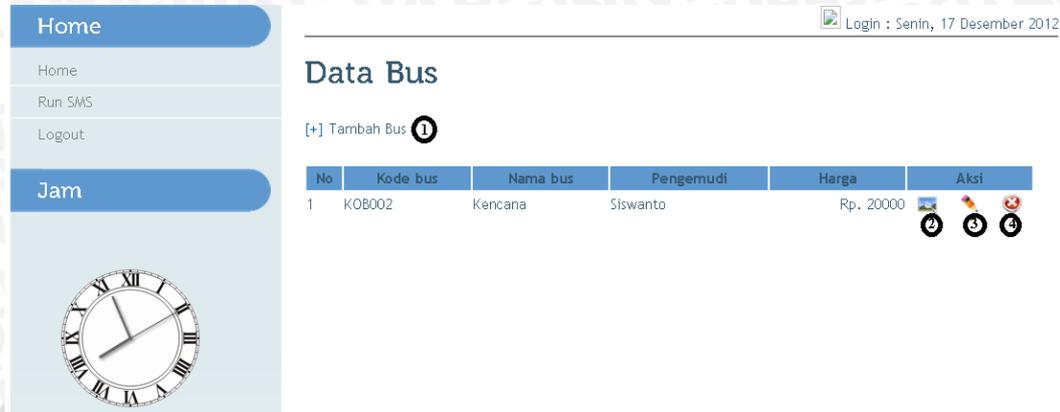
**Gambar 4.58 Tampilan Antarmuka Halaman Pengemudi**  
Sumber : [Perancangan]

Gambar 4.58 memiliki keterangan sebagai berikut :

1. Tombol Tambah Pengemudi untuk menambah data pengemudi.
2. Tombol Edit untuk mengubah data pengemudi.
3. Tombol Hapus untuk menghapus data pengemudi.

e. **Halaman Bus**

Halaman bus berfungsi bagi admin untuk menambah, melihat, mengubah dan menghapus data bus.



**Gambar 4.59 Tampilan Antarmuka Halaman Bus**  
Sumber : [perancangan]

Gambar 4.59 memiliki keterangan sebagai berikut :

1. Tombol Tambah Bus untuk menambah data bus.
2. Tombol Gambar untuk melihat data bus.
3. Tombol Edit untuk mengubah data bus.
4. Tombol *Hapus* untuk menghapus data bus.

f. **Halaman Jadwal**

Halaman Jadwal berfungsi bagi admin untuk melihat jadwal keberangkatan bus pada tanggal pada tanggal itu saja dan menambah jadwal keberangkatan bus.



**Gambar 4.60 Tampilan Antarmuka Halaman Jadwal**  
Sumber : [Perancangan]

Gambar 4.60 memiliki keterangan sebagai berikut :

1. Menu Tambah Jadwal untuk menambah jadwal keberangkatan bus.
2. Menu Tambah Penjadwalan untuk menyesuaikan bus dengan jadwal yang telah dibuat.

**g. Halaman Pemesanan**

Halaman Pemesanan berfungsi bagi admin untuk mengkonfirmasi data pemesanan tiket calon penumpang.



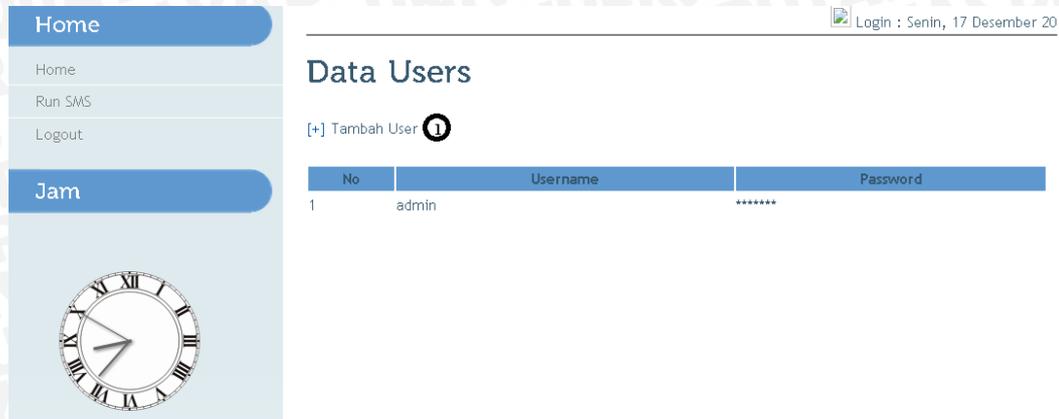
**Gambar 4.61 Tampilan Antarmuka Halaman Pemesanan**  
Sumber : [Perancangan]

Gambar 4.61 memiliki keterangan sebagai berikut :

1. Tombol Pemesanan Belum Dikonfirmasi untuk melihat data pesanan yang belum dikonfirmasi.
2. Tombol Pemesanan Sudah Dikonfirmasi untuk melihat data pesanan yang sudah dikonfirmasi.
3. Tombol Pemesanan Pemesanan Ditolak untuk melihat data pesanan *user* yang ditolak.
4. Kolom Status Konfirmasi untuk mengubah status konfirmasi pemesanan.
5. Tombol Gambar untuk melihat data pemesanan.
6. Tombol *Update* untuk mengupdate status konfirmasi.

## h. Halaman Users

Halaman users berfungsi bagi admin untuk menambah data admin.

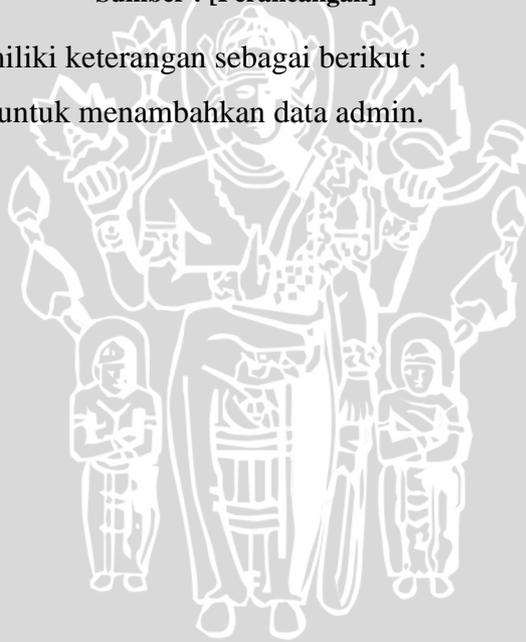


No	Username	Password
1	admin	*****

**Gambar 4.62** Tampilan Antarmuka Halaman Users  
Sumber : [Perancangan]

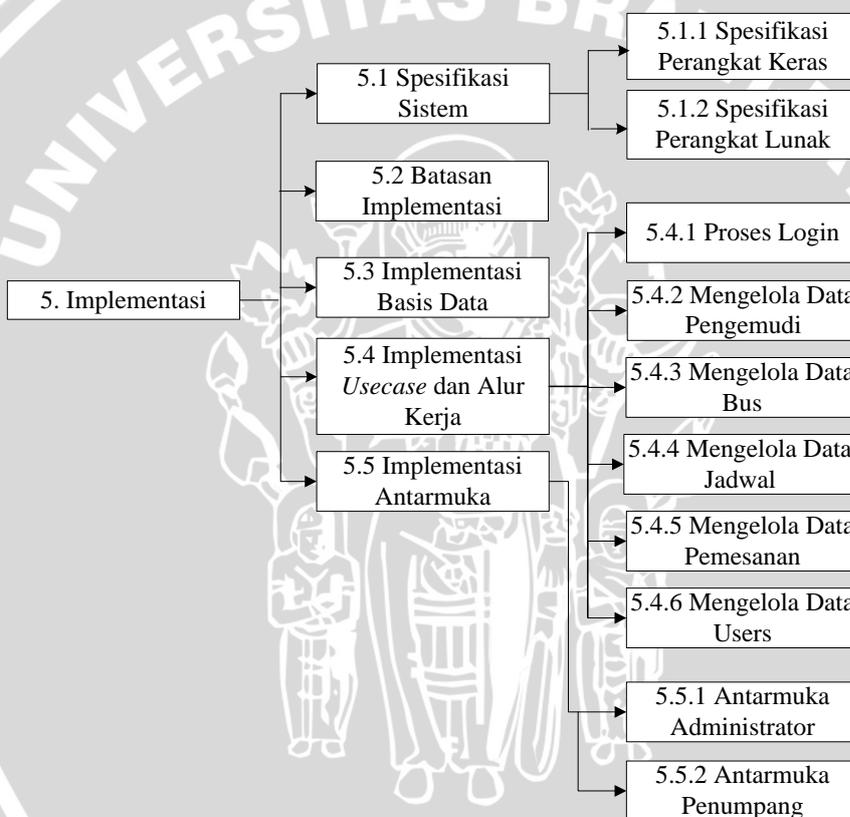
Gambar 4.62 memiliki keterangan sebagai berikut :

Tombol Tambah User untuk menambahkan data admin.



## BAB V IMPLEMENTASI

Bab ini membahas mengenai tahapan implementasi perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang 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 5.1 Pohon Implementasi**

Sumber : [Implementasi]

### 5.1 SPESIFIKASI PLATFORM PENGEMBANGAN

Perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang dikembangkan dalam lingkungan implementasi yang terdiri dari perangkat keras dan perangkat lunak.

### 5.1.1. Spesifikasi Lingkungan Perangkat Keras

Spesifikasi perangkat keras yang dipakai dalam proses pengembangan dijelaskan pada Tabel 5.1.

**Tabel 5.1 Spesifikasi perangkat keras komputer**

<b>Sony Vaio VGN CR-353</b>	
<i>Processor</i>	Intel (R) Core (TM) 2 Duo CPU T8100 2.10 GHz
<i>Memory (RAM)</i>	2.00 GB
<i>Harddisk</i>	Hard Disk Drive 250 GB
<i>Motherboard</i>	Dell Note Intel Motherboard
<i>Graphic Card</i>	ATI Mobility Radeon X2300 128MB dedicated DirectX 9.0 SM 3.0
<i>Monitor</i>	14.1" WXGA Display
<b>Huawei Mobile Connect Modem</b>	
<i>Model</i>	E160
<i>Jenis</i>	HSDPA USB <i>Stick</i>

Sumber : [Implementasi]

### 5.1.2. Spesifikasi Lingkungan Perangkat Lunak

Spesifikasi perangkat lunak yang dipakai dalam proses pengembangan Perangkat lunak SMS *Gateway* pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang dijelaskan pada Tabel 5.2.

**Tabel 5.2 Spesifikasi perangkat lunak komputer**

<b>Sony Vaio VGN CR-353</b>	
<i>Operating System</i>	Microsoft Windows 7 32-bit
<i>Programming Language</i>	PHP
<i>Database Management System</i>	MySQL 5.0.7
<i>Integrated Development Environment</i>	Macromedia Dreamwaver, Wordpad
<i>Software Development Tools</i>	Gammu 0.4, XAMPP

Sumber : [Implementasi]

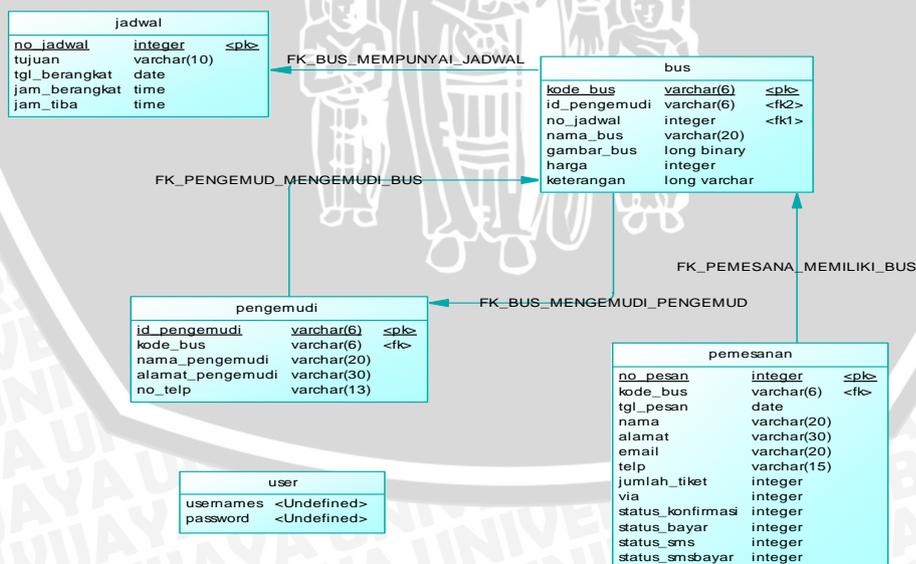
## 5.2 BATASAN – BATASAN IMPLEMENTASI

Beberapa batasan dalam mengimplementasikan perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang sebagai berikut :

1. Perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang dirancang dan dijalankan dengan menggunakan Website dan aplikasi SMS.
2. Database Management System yang digunakan adalah MySQL.
3. Perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang menggunakan tarif SMS dan nomor handphone regular.
4. Perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang menggunakan 1 basis data server pada komputer admin.

### 5.3 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 5.2 menggambarkan diagram konseptual entity relationship dari Perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang .



Gambar 5.2 Diagram ER konseptual

Sumber : [Implementasi]



Berikut ini merupakan struktur tabel serta keterangan masing masing tabel dan *field* yang ada pada *database*. Tabel berisi *names* dan *password* yang dapat dimiliki pengguna aplikasi SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang. Struktur tabel admin ditunjukkan pada Tabel 5.3.

**Tabel 5.3 Struktur tabel admin**

No.	Nama <i>Field</i>	Tipe	Lebar	Keterangan
1	Username	Varchar	30	nama Admin
2	Password	Varchar	30	<i>Password</i> Admin

Sumber : [Perancangan]

Tabel jadwal berisi data jadwal bus. Data jadwal tersebut meliputi nomor jadwal, tujuan, tanggal berangkat, jam berangkat, jam tiba. Struktur tabel jadwal ditunjukkan pada Tabel 5.4.

**Tabel 5.4. Struktur tabel jadwal**

No.	Nama <i>Field</i>	Tipe	Lebar	Keterangan
1	no_jadwal	Integer	11	No jadwal bus
2	Tujuan	Varchar	10	Tujuan bus
3	tgl_berangkat	Date		Tanggal keberangkatan bus
4	jam_berangkat	Time		Jam keberangkatan bus
5	jam_tiba	Time		Jam kedatangan bus

Sumber : [Perancangan]

Data pengemudi tersebut meliputi *id* pengemudi, kode bus, nama pengemudi, alamat pengemudi dan nomor telepon. Struktur tabel antrian ditunjukkan pada Tabel 5.5.

**Tabel 5.5. Struktur tabel pengemudi**

No.	Nama <i>Field</i>	Tipe	Lebar	Keterangan
1	id_pengemudi	Varchar	6	<i>Id</i> pengemudi
2	kode_bus	Varchar	6	Kode bus
3	nama_pengemudi	Varchar	20	Nama pengemudi
4	alamat_pengemudi	Varchar	30	Alamat pengemudi
5	no_telp	Varchar	13	No telepon pengemudi

Sumber : [Perancangan]

Data bus tersebut meliputi kode bus, nomor jadwal, *id* pengemudi, nama bus, gambar bus, harga, keterangan. Struktur tabel bus ditunjukkan pada Tabel 5.6.

**Tabel 5.6. Struktur tabel bus**

No.	Nama Field	Tipe	Lebar	Keterangan
1	kode_bus	Varchar	6	Kode bus
2	no_jadwal	Integer	11	No jadwal bus
3	<i>id</i> _pengemudi	Varchar	6	<i>Id</i> _pengemudi
4	nama_bus	Varchar	20	Nama bus
5	gambar_bus	Longblob		Gambar bus
6	Harga	Integer	11	Harga tiket bus
7	Keterangan	text		Keterangan

Sumber : [Perancangan]

Data pemesanan meliputi nomor pesan, tanggal pesan, kode bus, nama, alamat, email, telepon, jumlah tiket, via, status konfirmasi, status bayar, status sms dan status smsbayar. Struktur pemesanan ditunjukkan pada Tabel 5.7.

**Tabel 5.7. Struktur tabel pesan**

No.	Nama Field	Tipe	Lebar	Keterangan
1	no_pesan	Integer	11	Nomor pemesanan
2	tgl_pesan	Date		Tanggal pemesanan
3	kode_bus	Varchar	6	Kode bus
4	Nama	Varchar	20	Nama pesan
5	Alamat	Varchar	30	Alamat pesan
6	Email	Varchar	20	Email pesan
7	Telp	Integer	12	Telepon pesan
8	jumlah_tiket	Integer	2	Jumlah tiket pesan
9	Via	Integer	1	Pemesanan via
10	status_konfirmasi	Integer	1	Status konfirmasi pemesanan
11	Status_bayar	Integer	1	Status pembayaran

12	Status_sms	Integer	1	Status sms
13	Status_smsbayar	Integer	1	Status konfirmasi sms pembayaran

Sumber : [Perancangan]

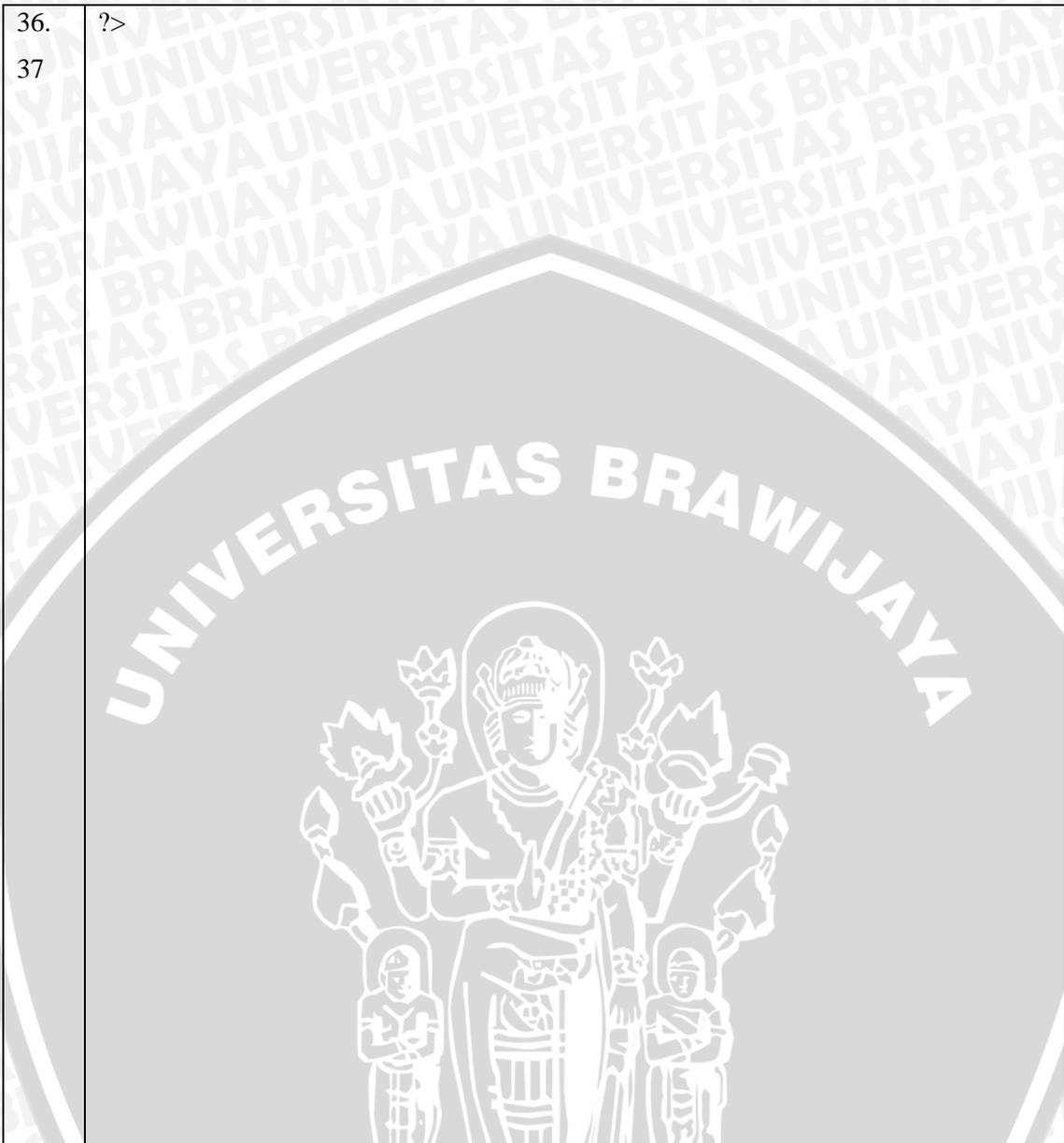
## 5.4 IMPLEMENTASI USE CASE DAN ALUR KERJA

Aplikasi SMS Gateway pada Sistem Reservasi Tiket mempunyai beberapa proses utama yang terbagi dalam beberapa *file PHP*. Pada penulisan skripsi ini hanya dicantumkan algoritma dari beberapa proses saja sehingga tidak semua implementasi algoritma proses akan dicantumkan. Implementasi algoritma proses yang dicantumkan antara lain adalah proses halaman masuk, mengelola pengemudi, mengelola bus, dan mengelola jadwal mengelola pemesanan dan mengelola user. Implementasi algoritma ini akan direpresentasikan dalam bentuk *code* dengan bahasa pemrograman *PHP*.

### 5.4.1 Implementasi Proses Login

Pengguna dapat mengisi *field username* dan *password* yang telah tersedia di halaman *desktop* perangkat lunak SMS Gateway pada Sistem Reservasi Tiket. Gambar 5.3 merupakan proses *cek\_login* yang mengimplementasikan Perancangan Alur kerja Admin Login.

```
1. <?php
2. include "../config/koneksi.php";
3. function antiinjection($data){
4.     $filter_sql =
5.         mysql_real_escape_string(stripslashes(strip_tags(htmlspecialchars($data,ENT
6.             _QUOTES))));
7.     return $filter_sql;
8. }
9.
10. $username = antiinjection($_POST['username']);
11. $pass = antiinjection($_POST['password']);
12.
13. $login=mysql_query("SELECT * FROM `user` WHERE `usernames`='$username'
14.     AND `password`='$pass'");
15. $ketemu=mysql_num_rows($login);
16. $r=mysql_fetch_array($login);
17.
18. // Apabila username dan password ditemukan
19. if ($ketemu > 0){
20.     session_start();
21.
22.     $_SESSION['user'] = $r['usernames'];
23.     $_SESSION['pass'] = $r['password'];
24.
25.     header('location:media.php?module=home');
26. }
27. else{
28.     echo"<script>alert('LOGIN GAGAL! Username atau Password Anda tidak
29.     benar')</script>";
30.
31.     print("<html><head><meta http-equiv='refresh'
32.     content='0;url=index.php'></head><body></body></html>");
33.     exit;
34.
35. }
```



Gambar 5.3 Implementasi Proses Login

Sumber : [Implementasi]

Penjelasan implementasi algoritma Halaman Masuk pada Gambar 5.3 yaitu:

1. Baris 2 merupakan proses koneksi *database*.
2. Baris 3-8 merupakan fungsi *anti SQL Injection*.
3. Baris 10-11 merupakan proses pengambilan data masukan dari halaman *login*.

4. Baris 13-16 merupakan proses pengecekan status dalam basis data, sehingga aplikasi yang dibuka sesuai dengan status username dan password yang tersimpan dalam basis data.
5. Baris 18-26 merupakan proses pengarahannya halaman jika pengecekan sesuai.
6. Baris 27-37 merupakan proses penampilan pesan kesalahan jika pengecekan tidak sesuai.

#### 5.4.2 Implementasi Proses Mengelola Data Pengemudi

Pengguna yang terdaftar sebagai admin dapat menambah, mengubah, dan menghapus data pengemudi yang digunakan aplikasi SMS Gateway pada Sistem Reservasi Tiket ini. Gambar 5.4 merupakan proses mengelola data pengemudi yang mengimplementasikan Perancangan Alur kerja Admin Mengelola Data Pengemudi Pengemudi.

```

1. <?php
2. include "../././config/authentication_users.php";
3. include "../././config/koneksi.php";
4.
5. $module=$_GET['module'];
6. $act=$_GET['act'];
7.
8. if ($module == 'pengemudi' && $act == 'input'){
9.   if ($_POST['kode'] == "") {
10.     echo"<script>alert('masukkan id pengemudi terlebih dahulu.')</script>";
11.     echo"<script>document.location.href='javascript:history.go(-1)';</script>";
12.
13.   }
14.   else if(($_POST['nama'] == "") || ($_POST['alamat'] == "") || ($_POST['notelp'] ==
15.     "")) {
16.     echo"<script>alert('data masih ada yang kosong.')</script>";
17.     echo"<script>document.location.href='javascript:history.go(-1)';</script>";
18.
19.   }

```

```
20. else {
21.     $cek = mysql_query("SELECT * FROM `pengemudi` WHERE
22.         `id_pengemudi`='$_POST[kode]");
23.     $stampil = mysql_num_rows($cek);
24.
25.     if ($stampil > 0){
26.         echo"<script>alert('id pengemudi sudah digunakan! gunakan id pengemudi
27.         yang lain.')</script>";
28.         echo"<script>document.location.href='javascript:history.go(-1)';</script>";
29.     }
30.     else{
31.         mysql_query("INSERT INTO pengemudi(id_pengemudi,
32.                                             kode_bus,
33.                                             nama_pengemudi,
34.                                             alamat_pengemudi,
35.                                             no_telp)
36.                     VALUES('$_POST[kode]',
37.                               'NULL',
38.                               '$_POST[nama]',
39.                               '$_POST[alamat]',
40.                               '$_POST[notelp]");
41.
42.         header('location:../../media.php?module='.$module);
43.     }
44. }
45. }
46.
47.
48. else if ($module=='pengemudi' && $act=='delete'){
49.     mysql_query("DELETE FROM pengemudi WHERE id_pengemudi =
50.         '$_GET[kode]");
51.     if ($_GET['bus'] != "NULL") {
52.         mysql_query("UPDATE bus SET id_pengemudi = 'NULL'
53.             WHERE kode_bus = '$_GET[bus]");
54.     }
```

```
55. header('location:../../media.php?module='.$module);
56. }
57.
58.
59. else if ($module=='pengemudi' && $act=='update'){
60.     mysql_query("UPDATE pengemudi SET nama_pengemudi =
61.     '$_POST[nama]',
62.
63.     alamat_pengemudi = '$_POST[alamat]',
64.     no_telp =
65.     '$_POST[notelp]'
66.     WHERE id_pengemudi = '$_POST[kode]'");
67.     header('location:../../media.php?module='.$module);
68. }
69.
70.
71. ?>
```

Gambar 5.4 Implementasi Proses Mengelola Data Pengemudi

Sumber : [Implementasi]

Penjelasan implementasi algoritma Mengelola Data Pengemudi pada Gambar

5.4 yaitu:

1. Baris 2-3 merupakan proses koneksi *database*.
2. Baris 5-21 merupakan proses pengambilan data masukan.
3. Baris 22-25 merupakan proses penampilan data pengemudi.

4. Baris 27-53 merupakan proses penambahan data pengemudi.
5. Baris 56-65 merupakan proses penghapusan data pengemudi.
6. Baris 68-81 merupakan proses perubahan data pengemudi.

### 5.4.3 Implementasi Proses Mengelola Data Bus

Pengguna yang terdaftar sebagai admin dapat melihat, menambah, dan menghapus data bus yang digunakan aplikasi SMS Gateway pada Sistem Reservasi Tiket ini. Gambar 5.5 merupakan proses mengelola data bus yang mengimplementasikan Perancangan Alur kerja Admin Mengelola Data Bus.

```

1.      <?php
2.      include ".././././config/authentication_users.php";
3.      include ".././././config/koneksi.php";
4.
5.      $module=$_GET['module'];
6.      $act=$_GET['act'];
7.
8.      //tambah bus
9.      if ($module == 'bus' && $act == 'input'){
10.         if ($_POST['kode'] == "") {
11.             echo"<script>alert('masukkan kode bus terlebih dahulu.')</script>";
12.             echo"<script>document.location.href='javascript:history.go(-1)';</script>";
13.
14.         }
15.         else if(($_POST['nama'] == "") || ($_POST['pengemudi'] == "0") ||
16.             ($_POST['harga'] == "") || ($_POST['keterangan'] == "")) {
17.             echo"<script>alert('data masih ada yang kosong.')</script>";
18.             echo"<script>document.location.href='javascript:history.go(-1)';</script>";
19.
20.         }
21.         else {
22.             $cek = mysql_query("SELECT * FROM `bus` WHERE
23.                 `kode_bus`='".$_POST[kode]'");
24.             $stampil = mysql_num_rows($cek);

```

```
25.
26.     if ($stampil > 0){
27.         echo"<script>alert('kode bus sudah digunakan! gunakan kode bus yang
28.             lain.')</script>";
29.         echo"<script>document.location.href='javascript:history.go(-1)';</script>";
30.     }
31.     else{
32.         $file_data = $_FILES['file_data']['tmp_name'];
33.         $data = addslashes(fread(fopen($file_data, "r"), filesize($file_data)));
34.
35.         mysql_query("INSERT INTO bus(kode_bus,
36.                                     no_jadwal,
37.                                     id_pengemudi,
38.                                     nama_bus,
39.                                     gambar_bus,
40.                                     harga,
41.                                     keterangan)
42.                     VALUES('$_POST[kode]',
43.                               'NULL',
44.                               '$_POST[pengemudi]',
45.                               '$_POST[nama]',
46.                               '$data',
47.                               '$_POST[harga]',
48.                               '$_POST[keterangan]");
49.
50.         mysql_query("UPDATE pengemudi SET kode_bus = '$_POST[kode]'
51.                     WHERE id_pengemudi = '$_POST[pengemudi]");
52.         header('location:../../media.php?module='.$module);
53.     }
54. }
55. }
56.
57. //hapus bus
58. else if ($module=='bus' && $act=='delete'){
59.     mysql_query("DELETE FROM bus WHERE kode_bus = '$_GET[kode]'");
```

```
60.     mysql_query("UPDATE pengemudi SET kode_bus = 'NULL'
61.           WHERE id_pengemudi = '$_GET[pengemudi]'");
62.     header('location:../../media.php?module='.$module);
63.     }
64.
65.     //update bus
66.     else if ($module=='bus' && $act=='update'){
67.         $file_data=$_FILES['file_data']['tmp_name'];
68.         if (($file_data) && ($_POST['pengemudi'] != "0")) {
69.             $data = addslashes(fread(fopen($file_data, "r"), filesize($file_data)));
70.             mysql_query("UPDATE bus SET id_pengemudi = '$_POST[pengemudi]',
71.                       nama_bus =
72.                       '$_POST[nama]',
73.                       gambar_bus =
74.                       '$data',
75.                       harga =
76.                       '$_POST[harga]',
77.                       keterangan =
78.                       '$_POST[keterangan]'
79.             WHERE kode_bus = '$_POST[kode]'");
80.             mysql_query("UPDATE pengemudi SET kode_bus = '$_POST[kode]'
81.             WHERE id_pengemudi = '$_POST[pengemudi]'");
82.         }
83.     else if (($file_data) && ($_POST['pengemudi'] == "0")){
84.         $data = addslashes(fread(fopen($file_data, "r"), filesize($file_data)));
85.         mysql_query("UPDATE bus SET nama_bus = '$_POST[nama]',
86.                   gambar_bus =
87.                   '$data',
88.                   harga =
89.                   '$_POST[harga]',
90.                   keterangan =
91.                   '$_POST[keterangan]'
92.         WHERE kode_bus = '$_POST[kode]'");
93.     }
94.     else if ((!$file_data) && ($_POST['pengemudi'] != "0")) {
```

```
95.      mysql_query("UPDATE bus SET id_pengemudi = '$_POST[pengemudi]',
96.                                     nama_bus =
97.      '$_POST[nama]',
98.                                     harga =
99.      '$_POST[harga]',
100.                                    keterangan =
101.      '$_POST[keterangan]'
102.      WHERE kode_bus = '$_POST[kode]");
103.      mysql_query("UPDATE pengemudi SET kode_bus = '$_POST[kode]'
104.      WHERE id_pengemudi = '$_POST[pengemudi]");
105.      }
106.      else {
107.      mysql_query("UPDATE bus SET nama_bus = '$_POST[nama]',
108.                                     harga =
109.      '$_POST[harga]',
110.                                     keterangan =
111.      '$_POST[keterangan]'
112.      WHERE kode_bus = '$_POST[kode]");
113.      }
114.      header('location:../../media.php?module='.$module);
115.      }
116.
117.
118.      ?>
```

130.
131.
132.
133.
134.
135.
136.
137.
138.
139.
140.
141.
142.
143.
144.
145.
146.
147.
148.
149.
150.
151.

**Gambar 5.5 Implementasi Alur kerja Admin Mengelola Data Bus**

Sumber : [Implementasi]

Penjelasan implementasi algoritma Mengelola Data Bus Aplikasi Admin pada Gambar 5.5 yaitu:

1. Baris 2-3 merupakan proses koneksi *database*.
2. Baris 5-67 merupakan proses penambahan data bus.
3. Baris 69-80 merupakan proses penghapusan data bus.
4. Baris 82-148 merupakan proses perubahan data bus.

#### 5.4.4 Implementasi Proses Mengelola Data Jadwal Bus

Pengguna yang terdaftar sebagai admin dapat melihat dan menambah data jadwal yang digunakan aplikasi SMS Gateway pada Sistem Reservasi Tiket ini. Gambar 5.6 merupakan proses mengelola data bus yang mengimplementasikan Perancangan Alur kerja Admin Mengelola Data Jadwal Bus.

```
1. <?php
2. include ".././././config/authentication_users.php";
3. include ".././././config/koneksi.php";
4.
5. $module=$_GET['module'];
6. $act=$_GET['act'];
7.
8. if ($module=='jadwal' && $act=='input'){
9.     if ($_POST['tujuan'] == "0") {
10.         echo"<script>alert('pilih tujuan terlebih dahulu.')</script>";
11.         echo"<script>document.location.href='javascript:history.go(-1)';</script>";
12.
13.     }
14.     else {
15.         $cek = mysql_query("SELECT * FROM `jadwal` WHERE
16. `tujuan`='$ _POST[tujuan]' AND `tgl_berangkat`='$ _POST[tgl_berangkat]' AND
17. `jam_berangkat`='$ _POST[jam_berangkat]'
18. AND `jam_tiba`='$ _POST[jam_tiba]'");
19.         $stampil = mysql_num_rows($cek);
20.
21.         if ($stampil > 0){
22.             echo"<script>alert('jadwal sudah ada! buat jadwal lain.')</script>";
23.             echo"<script>document.location.href='javascript:history.go(-1)';</script>";
24.         }
25.         else{
26.             mysql_query("INSERT INTO jadwal(no_jadwal,
27.                                     tujuan,
28.                                     tgl_berangkat,
29.                                     jam_berangkat,
```

```
30.                                     jam_tiba)
31.                                     VALUES(NULL',
32.                                     '$_POST[tujuan]',
33.                                     '$_POST[tgl_berangkat]',
34.                                     '$_POST[jam_berangkat]',
35.                                     '$_POST[jam_tiba]');
36.
37.
38.                                     header('location:../../media.php?module='.$module);
39.                                     }
40.                                     }
41.                                     }
42.
43.     else if ($module=='jadwal' && $act=='inputjadwal'){
44.         if ($_POST['kode'] == "0") {
45.             echo"<script>alert('pilih kode bus terlebih dahulu.')</script>";
46.             echo"<script>document.location.href='javascript:history.go(-1)';</script>";
47.
48.         }
49.         else if ($_POST['nojadwal'] == "0") {
50.             echo"<script>alert('pilih no jadwal terlebih dahulu.')</script>";
51.             echo"<script>document.location.href='javascript:history.go(-1)';</script>";
52.
53.         }
54.         else {
55.             mysql_query("UPDATE bus SET no_jadwal = '$_POST[nojadwal]'
56.                 WHERE kode_bus = '$_POST[kode]'");
57.
58.             header('location:../../media.php?module='.$module);
59.         }
60.     }
61.
62. ?>
63.
64.
```

65.	
66.	
67.	
68.	
69.	
70.	
70.	
71.	
72.	
73.	
74.	

**Gambar 5.6 Implementasi Alur kerja Admin Mengelola Jadwal Bus**

Sumber : [Implementasi]

Penjelasan implementasi algoritma Mengelola Jadwal Bus Aplikasi Admin pada Gambar 5.6 yaitu:

1. Baris 2-3 merupakan proses koneksi *database*.
2. Baris 5-50 merupakan proses penambahan data jadwal bus.
3. Baris 53-74 merupakan proses penambahan jadwal bus pada bus yang terkait.

#### 5.4.5 Implementasi Proses Mengelola Data Pemesanan

Pengguna yang terdaftar sebagai admin dapat melihat dan mengkonfirmasi data pemesanan yang digunakan aplikasi SMS Gateway pada Sistem Reservasi Tiket ini. Gambar 5.7 merupakan proses mengelola data pemesanan yang mengimplementasikan Perancangan Alur kerja Admin Mengelola Data Pemesanan.

1.	<?php
2.	include "../././config/authentication_users.php";
3.	include "../././config/koneksi.php";
4.	
5.	\$module=\$_GET['module'];
6.	\$act=\$_GET['act'];



```
7.
8.   if ($module=='pemesanan' && $act=='updatestatus'){
9.       $jmldata = $_POST['jmldata'];
10.
11.      for ($i=0; $i<$jmldata; $i++) {
12.          $status[$i] = $_POST['status'.$i];
13.          $no_pesanan[$i] = $_POST['nopesanan'.$i];
14.
15.          mysql_query("UPDATE pemesanan SET status_konfirmasi = '$status[$i]'
16.                      WHERE no_pesanan = '$no_pesanan[$i]'");
17.      }
18.      header('location:../../media.php?module='.$module);
19.  }
20.
21.  else if ($module=='pemesanan' && $act=='updatestatusbayar'){
22.      $jmldata = $_POST['jmldata'];
23.
24.      for ($i=0; $i<$jmldata; $i++) {
25.          $status[$i] = $_POST['verifikasi'.$i];
26.          $no_pesanan[$i] = $_POST['nopesanan'.$i];
27.
28.          if ($status[$i] != "") {
29.              mysql_query("UPDATE pemesanan SET status_bayar = '$status[$i]'
30.                          WHERE no_pesanan = '$no_pesanan[$i]'");
31.          }
32.      }
33.      header('location:../../media.php?module='.$module);
34.  }
35.
36.  ?>
```

**Gambar 5.7 Implementasi Alur kerja Admin Mengelola Data Pemesanan**

Sumber : [Implementasi]

Penjelasan implementasi algoritma Mengelola Data Pemesanan Aplikasi Admin pada Gambar 5.7 yaitu:

1. Baris 2-3 merupakan proses koneksi *database*.
2. Baris 5-21 merupakan proses perubahan status pemesanan pada data pemesanan.
3. Baris 23-41 merupakan proses perubahan status konfirmasi bayar pada data pemesanan.

## **5.5 IMPLEMENTASI ANTARMUKA APLIKASI**

Antarmuka Aplikasi SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang digunakan oleh pengguna untuk berinteraksi dengan sistem perangkat lunak.

### **5.5.1 Implementasi Antarmuka Admin**

Antarmuka pengguna untuk aplikasi admin berupa sebuah halaman *website*. Antarmuka aplikasi admin terdiri dari halaman *Login*, halaman *Home*, halaman *Run SMS*, keluar, halaman Pengemudi, halaman Bus, halaman Jadwal, halaman Pemesanan dan halaman *Users*.

#### **a. Halaman Login**

Halaman login merupakan salah satu antarmuka pengguna untuk aplikasi admin. Halaman *login* berfungsi mempermudah pengguna yang terdaftar sebagai admin untuk mengelola aplikasi ini. Gambar 4.14 akan menunjukkan perancangan tampilan antarmuka dari halaman *Login*.



Gambar 5.8 Tampilan Antarmuka Halaman *Login*

Sumber : [Implementasi]

**b. Halaman *Home***

Halaman *home* berfungsi bagi admin untuk mengelola *content website* dan mengelola akun admin.



Gambar 5.9 Tampilan Antarmuka Halaman Utama

Sumber : [Implementasi]

**c. Halaman *Run SMS***

Halaman *run SMS* berfungsi bagi admin untuk menjalankan program SMS GATEWAY.



Gambar 5.10 Tampilan Antarmuka Halaman *Run SMS*

Sumber : [Implementasi]

#### d. Halaman Pengemudi

Halaman pengemudi berfungsi bagi admin untuk menambah, mengubah dan menghapus data pengemudi.



No	Id Pengemudi	Nama Pengemudi	No Telp	Aksi
1	IP001	Deta	08564496611	[Edit] [Hapus]
2	IP002	Restu	08564561936	[Edit] [Hapus]
3	IP003	Dimas	08563568417	[Edit] [Hapus]
4	IP004	Anza	08990340093	[Edit] [Hapus]
5	IP005	Bias	085736030216	[Edit] [Hapus]
6	IP006	Mahardeka	085646411161	[Edit] [Hapus]
7	IP007	Wisnu	081805062617	[Edit] [Hapus]
8	IP008	Irwan	085236565410	[Edit] [Hapus]
9	IP009	Lukman	081917039174	[Edit] [Hapus]
10	IP010	Aldian	085646726507	[Edit] [Hapus]
11	IP011	Denny	085649904321	[Edit] [Hapus]
12	IP012	Kukuh	085646412251	[Edit] [Hapus]
13	IP013	Komang	085646327548	[Edit] [Hapus]
14	IP014	Faizal	081805037141	[Edit] [Hapus]
15	IP015	Walhyu	081803827728	[Edit] [Hapus]
16	IP016	Fabrian	081803020500	[Edit] [Hapus]
17	IP017	Prayoga	08886541013	[Edit] [Hapus]
18	IP018	Buce	085646596965	[Edit] [Hapus]
19	IP019	Adityo	0888371745	[Edit] [Hapus]
20	IP020	Lutfi	08563303030	[Edit] [Hapus]

Gambar 5.11 Tampilan Antarmuka Halaman Pengemudi

Sumber : [Implementasi]

#### e. Halaman Bus

Halaman bus berfungsi bagi admin untuk menambah, melihat, mengubah dan menghapus data bus.



No	Kode bus	Nama bus	Pengemudi	Harga	Aksi
1	KB001	TiaraMas	Deta	Rp. 310000	[Edit] [Hapus]
2	KB002	ALS	Restu	Rp. 700000	[Edit] [Hapus]
3	KB003	Karina	Dimas	Rp. 330000	[Edit] [Hapus]
4	KB004	KramatJati	Anza	Rp. 180000	[Edit] [Hapus]
5	KB005	Handoyo	Bias	Rp. 105000	[Edit] [Hapus]
6	KB006	TittanMas	Mahardeka	Rp. 370000	[Edit] [Hapus]
7	KB007	Lorena	Wisnu	Rp. 260000	[Edit] [Hapus]
8	KB008	Zena	Irwan	Rp. 90000	[Edit] [Hapus]
9	KB009	Karina	Lukman	Rp. 650000	[Edit] [Hapus]
10	KB010	Kencana	Aldian	Rp. 260000	[Edit] [Hapus]
11	KB011	Handoyo	Denny	Rp. 92000	[Edit] [Hapus]
12	KB012	Coyo	Kukuh	Rp. 195000	[Edit] [Hapus]
13	KB013	Megah	Komang	Rp. 120000	[Edit] [Hapus]
14	KB014	TittanMas	Faizal	Rp. 360000	[Edit] [Hapus]
15	KB015	ALS	Walhyu	Rp. 510000	[Edit] [Hapus]
16	KB016	MallinPutra	Fabrian	Rp. 230000	[Edit] [Hapus]
17	KB017	Karina	Prayoga	Rp. 620000	[Edit] [Hapus]
18	KB018	SafariDharmaRaya	Buce	Rp. 110000	[Edit] [Hapus]

Gambar 5.12 Tampilan Antarmuka Halaman Bus

Sumber : [Implementasi]

#### f. Halaman Jadwal

Halaman Jadwal berfungsi bagi admin untuk melihat jadwal keberangkatan bus pada tanggal pada tanggal itu saja dan menambah jadwal keberangkatan bus.



Gambar 5.13Tampilan Antarmuka Halaman Jadwal

Sumber : [Implementasi]

g. **Halaman Pemesanan**

Halaman Pemesanan berfungsi bagi admin untuk mengkonfirmasi data pemesanan tiket calon penumpang.



Gambar 5.14 Tampilan Antarmuka Halaman Pemesanan

Sumber : [Implementasi]

## h. Halaman Users

Halaman users berfungsi bagi admin untuk menambah data admin.



**Gambar 5.15 Tampilan Antarmuka Halaman Users**

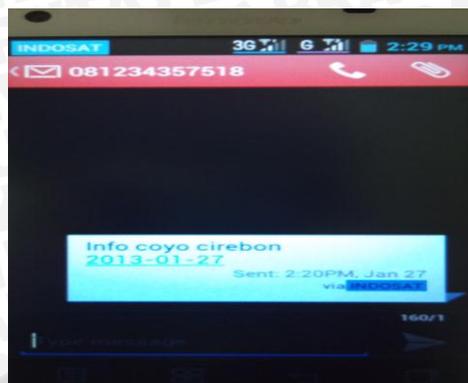
Sumber : [Implementasi]

## 5.5.2 Implementasi Antarmuka Penumpang

Antarmuka pengguna untuk aplikasi penumpang berupa sebuah format SMS sistem antrian berbasis SMS *Gateway*. Antarmuka aplikasi penumpang terdiri dari SMS Melihat Info, SMS Info Bus, SMS Pemesanan Tiket, SMS Konfirmasi Pemesanan Tiket Berhasil, SMS Konfirmasi Bayar.

### a. SMS Melihat Info

SMS Melihat Info merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. Format SMS ini berfungsi mempermudah penumpang untuk mengetahui info bus melalui SMS. Perancangan antarmuka ini mengacu pada spesifikasi kebutuhan SRS\_001\_01. Gambar 5.16 akan menunjukkan perancangan tampilan antarmuka dari SMS Melihat Info.



Gambar 5.16 Tampilan Antarmuka SMS Melihat Info

Sumber : [Perancangan]

**b. SMS Info Bus**

SMS Info Bus merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. SMS ini merupakan balasan dari sistem tentang informasi bus yang dicari oleh penumpang. Gambar 5.17 akan menunjukkan perancangan tampilan antarmuka dari SMS Info Bus.



Gambar 5.17 Tampilan Antarmuka SMS Informasi Bus

Sumber : [Perancangan]

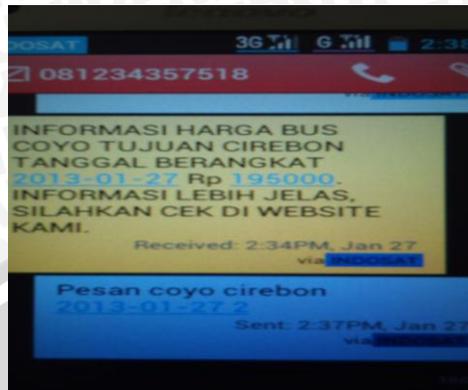
Gambar 5.17 menampilkan format SMS yang diterima penumpang, apabila bus yang dicari tersedia.

**c. SMS Pemesanan Tiket**

SMS Pemesanan Tiket merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. Format SMS ini berfungsi mempermudah penumpang untuk memesan tiket melalui SMS. Perancangan antarmuka ini mengacu pada

repository.ub.ac.id

spesifikasi kebutuhan SRS\_001\_02. Gambar 5.18 akan menunjukkan perancangan tampilan antarmuka dari SMS Memesan Tiket.

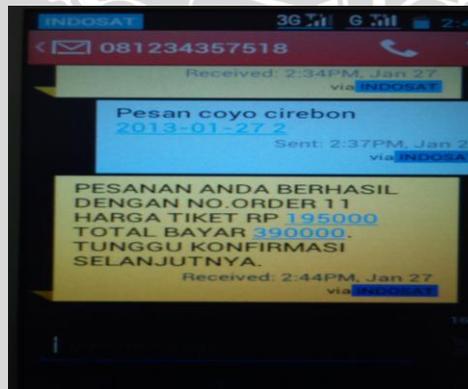


Gambar 5.18 Tampilan Antarmuka SMS Memesan Tiket

Sumber : [Perancangan]

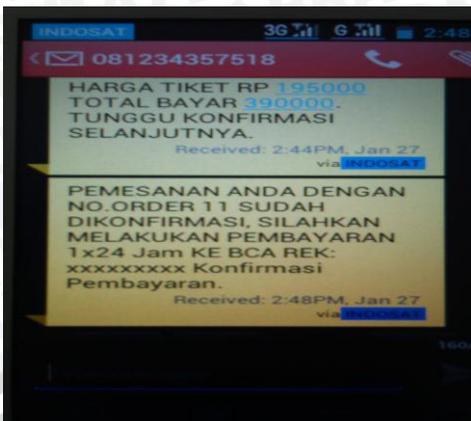
#### d. SMS Konfirmasi Pemesanan Tiket

SMS Konfirmasi Pemesanan Tiket merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. SMS ini merupakan balasan dari sistem tentang konfirmasi bus yang dipesan oleh penumpang. Gambar 5.19 akan menunjukkan perancangan tampilan antarmuka dari SMS Konfirmasi Pemesanan Tiket.



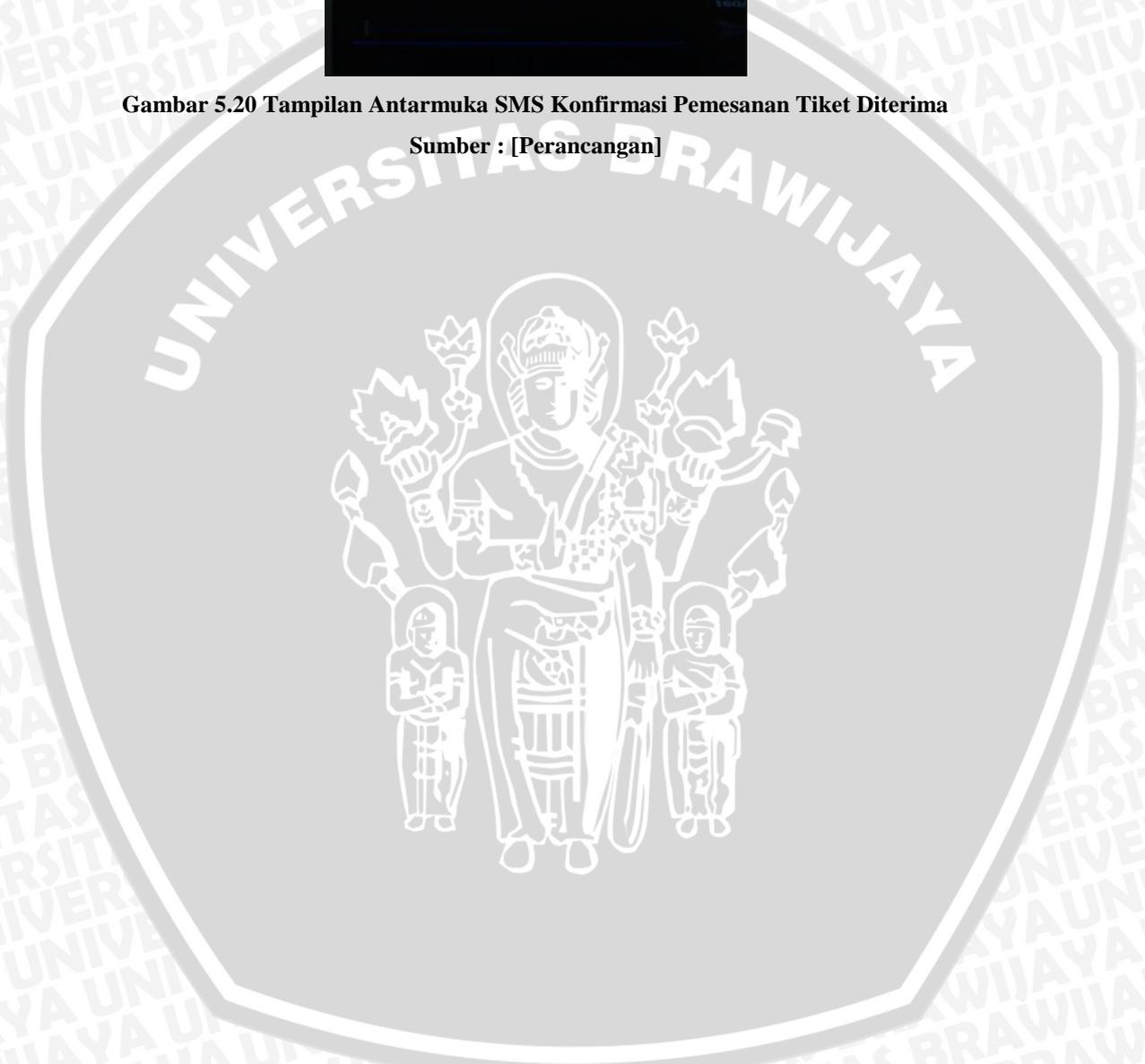
Gambar 5.19 Tampilan Antarmuka SMS Konfirmasi Pemesanan Tiket

Sumber : [Perancangan]



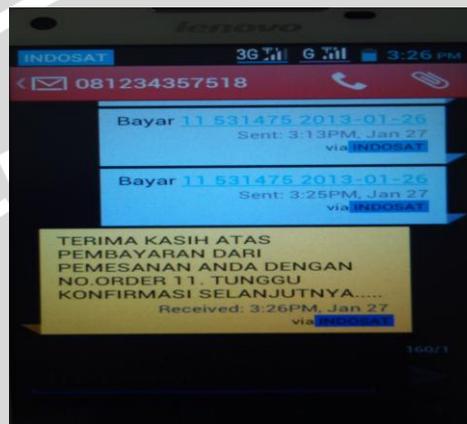
Gambar 5.20 Tampilan Antarmuka SMS Konfirmasi Pemesanan Tiket Diterima

Sumber : [Perancangan]



e. **SMS Konfirmasi Bayar**

SMS Konfirmasi Bayar merupakan salah satu antarmuka pengguna untuk aplikasi penumpang. Format SMS ini berfungsi mempermudah penumpang untuk mengkonfirmasi pembayaran pemesanan tiket melalui SMS. Perancangan antarmuka ini mengacu pada spesifikasi kebutuhan SRS\_001\_04. Gambar 5.21 akan menunjukkan perancangan tampilan antarmuka dari SMS Konfirmasi Bayar.



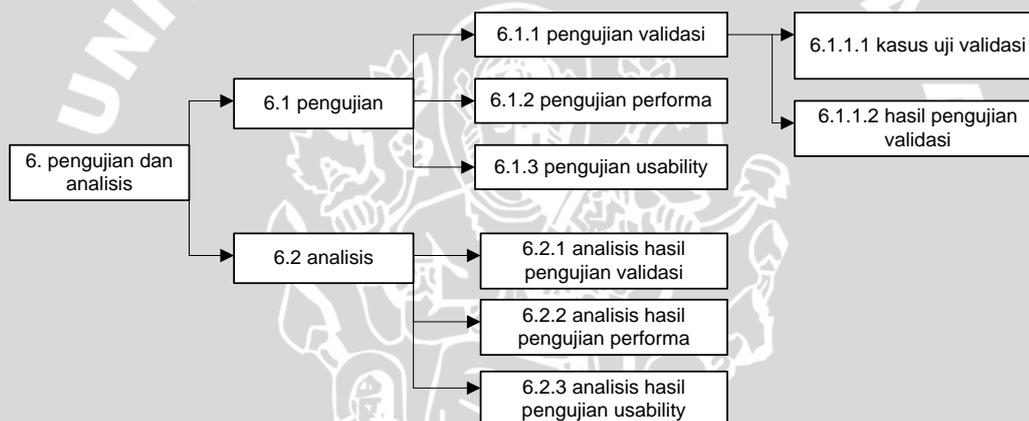
**Gambar 5.21 Tampilan Antarmuka SMS Konfirmasi Pembayaran**

**Sumber : [Perancangan]**

## BAB VI

### PENGUJIAN DAN ANALISIS

Pada bab ini dilakukan proses pengujian dan analisis terhadap aplikasi perangkat lunak SMS Gateway pada Sistem Reservasi Tiket Bus di Terminal Arjosari Malang yang telah dibangun. Proses pengujian dilakukan melalui tiga tahapan (strategi) yaitu pengujian validasi, pengujian performa aplikasi dan pengujian usability. Pada pengujian validasi digunakan teknik pengujian *Black Box (Black Box Testing)*. Pada pengujian performa dilakukan dengan menghitung waktu *auto-reply sms* menggunakan aplikasi perangkat lunak SMS Gateway pada Sistem Reservasi Tiket di Terminal Arjosari Malang.



**Gambar 6.1**Pohon Pengujian dan Analisis

Sumber : [Pengujian dan Analisis]

#### 6.1 PENGUJIAN

Proses pengujian dilakukan melalui tiga tahapan (strategi) yaitu pengujian pengujian validasi, pengujian performa dan pengujian usability aplikasi perangkat lunak SMS Gateway pada Sistem Reservasi Tiket di Terminal Arjosari Malang.

##### 6.1.1 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 menggunakan metode pengujian *Black Box*, untuk memenuhi kebutuhan fungsional yang terdapat pada daftar kebutuhan. Pada skripsi ini dilakukan pengujian validasi terhadap perangkat lunak SMS Gateway pada Sistem Reservasi Tiket di Terminal Arjosari Malang.

### 6.1.1.1 Kasus Uji Validasi

**Tabel 6.1 Kasus uji untuk pengujian validasi login sah**

Nama Kasus Uji	Kasus Uji Login Sah
Objek Uji	SRS_002_01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas <i>login</i> bagi <i>user</i> dan <i>admin</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. <i>Admin</i> membuka <i>website</i>.</li> <li>2. <i>Admin</i> memilih kolom <i>Admin</i> pada halaman utama <i>website</i>.</li> <li>3. <i>Form</i> login ditampilkan.</li> <li>4. <i>Admin</i> memasukkan <i>userid</i> dan <i>password</i> di dalam <i>form</i> login.</li> <li>5. <i>Admin</i> menekan tombol <i>Login</i>.</li> </ol>
Hasil yang diharapkan	Sistem dapat melakukan penyeleksian kondisi <i>login</i> pada <i>database</i> berdasar data yang dimasukkan dan jika penyeleksian kondisi ini benar, maka akan masuk ke sistem sesuai dengan hak aksesnya.

Sumber : [Pengujian dan Analisis]

**Tabel 6.2 Kasus uji untuk pengujian validasi login tidak sah**

Nama Kasus Uji	Kasus Uji Login Tidak Sah
Objek Uji	SRS_002_01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas <i>login</i> bagi <i>Admin</i> .

Prosedur Uji	<ol style="list-style-type: none"> <li>1. <i>Admin</i> membuka <i>website</i>.</li> <li>2. <i>Admin</i> memilih kolom <i>Admin</i> pada halaman utama <i>website</i>.</li> <li>3. <i>Form</i> login ditampilkan.</li> <li>4. <i>Admin</i> memasukkan <i>userid</i> dan <i>password</i> di dalam <i>form</i> login.</li> <li>5. <i>Admin</i> menekan tombol <i>Login</i>.</li> </ol>
Hasil yang diharapkan	Sistem dapat melakukan penyeleksian kondisi login pada <i>database</i> berdasar data yang dimasukkan dan jika penyeleksian kondisi ini salah, maka tidak akan masuk ke sistem dan aplikasi menampilkan pesan kesalahan.

Sumber : [Pengujian dan Analisis]

**Tabel 6.3 Kasus uji untuk pengujian validasi *logout***

Nama Kasus Uji	Kasus Uji <i>Logout</i>
Objek Uji	SRS_002_08
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas <i>logout</i> bagi <i>Admin</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan pada keadaansudah <i>login</i></li> <li>2. <i>Admin</i> menekan tombol <i>Logout</i>.</li> </ol>
Hasil yang diharapkan	Sistem melakukan penghapusan session yang aktif dan menampilkan halaman login.

Sumber : [Pengujian dan Analisis]

**Tabel 6.4 Kasus uji untuk pengujian validasi menjalankan service sms gateway**

Nama Kasus Uji	Kasus Uji Menjalankan <i>Service SMS GATEWAY</i>
Objek Uji	SRS_002_02
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas menjalankan <i>service SMS GATEWAY</i> bagi <i>Admin</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Aplikasi dijalankan pada keadaansudah <i>login</i></li> <li>2. <i>Admin</i> menekan tombol "Jalankan <i>Service</i>" pada halaman <i>Run SMS</i>.</li> <li>3. <i>Admin</i> memilih <i>service SMS GATEWAY</i> yang ingin dijalankan.</li> </ol>
Hasil yang diharapkan	Sistem menjalankan <i>service</i> sesuai <i>service</i> yang dipilih oleh <i>admin</i> .

Sumber : [Pengujian dan Analisis]

**Tabel 6.5 Kasus uji untuk pengujian validasi mengelola data pengemudi**

Nama Kasus Uji	Kasus Uji Mengelola Data Pengemudi
Objek Uji	SRS_002_03
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas halaman untuk mengelola data pengemudi.
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan pada keadaansudah <i>login</i>.</li> <li>2. Admin memilih kolompengemudi pada halaman utamasistem.</li> <li>3. Admin menekan tombol tambah pengemudi dan mengisi field id pengemudi, nama pengemudi, alamat, no telp dan menekan tombol simpan.</li> <li>4. Admin memilih data pengemudi yang akan diubah dan mengubah data field dan menekan tombol update.</li> <li>5. Admin memilih data pengemudi yang akan dihapus dan menekan tombol hapus.</li> </ol>
Hasil yang diharapkan	Sistem menampilkan halaman data pengemudi. Sistem juga dapat menambah, mengubah dan menghapus data pengemudi.

Sumber : [Pengujian dan Analisis]

Tabel 6.6 Kasus uji untuk pengujian validasi mengelola data bus

Nama Kasus Uji	Kasus Uji Mengelola Data Bus
Objek Uji	SRS_002_04
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas halaman untuk mengelola data bus
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Aplikasi dijalankan pada keadaansudah <i>login</i></li> <li>2. Admin memilih kolom pengemudi pada halaman utama sistem.</li> <li>3. Admin menekan tombol tambah bus dan mengisi field kode bus, nama bus, memilih pengemudi, harga bus, memilih gambar bus, keterangan dan menekan tombol simpan.</li> <li>4. Admin memilih data bus yang akan diubah dan mengubah data field dan menekan tombol update.</li> <li>5. Admin memilih data bus yang ingin dilihat lebih jelas dan menekan tombol view.</li> <li>6. Admin memilih data bus yang ingin dihapus dan</li> </ol>

	menekan tombol hapus.
Hasil yang diharapkan	Sistem menampilkan halaman data bus. Sistem juga dapat menambah, mengubah, melihat dan menghapus data bus.

Sumber : [Pengujian dan Analisis]

**Tabel 6.7 Kasus uji untuk pengujian validasi mengelola data jadwal**

Nama Kasus Uji	Kasus Uji Mengelola Data Jadwal Bus
Objek Uji	SRS_002_05
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas halaman untuk mengelola data jadwal bus
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Aplikasi dijalankan pada keadaan sudah <i>login</i></li> <li>2. Admin memilih kolom jadwal pada halaman utama sistem.</li> <li>3. Admin menekan tombol tambah jadwal dan mengisi field memilih tujuan, tanggal berangkat, jam berangkat, jam tiba dan menekan tombol simpan.</li> <li>4. Admin menekan tombol tambah penjadwalan dan mengisi field memilih kode bus, id jadwal dan menekan tombol simpan.</li> </ol>
Hasil yang diharapkan	Sistem menampilkan halaman data bus. Sistem juga dapat menambah data jadwal bus.

Sumber : [Pengujian dan Analisis]

**Tabel 6.8 Kasus uji untuk pengujian validasi mengelola data pemesanan**

Nama Kasus Uji	Kasus Uji Mengelola Data Pemesanan
Objek Uji	SRS_002_06
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas halaman untuk mengelola data pemesanan.
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan pada keadaan sudah <i>login</i></li> <li>2. Admin memilih kolom pemesanan pada halaman utama sistem.</li> <li>3. Admin memilih kolom pemesanan belum dikonfirmasi pada halaman data pemesanan, admin memilih status konfirmasi dan menekan tombol update.</li> <li>4. Admin memilih kolom pemesanan sudah dikonfirmasi pada halaman data pemesanan, admin memilih status pembayaran dan menekan tombol update.</li> </ol>

	5. Admin memilih kolom pemesanan ditolak pada halaman data pemesanan untuk melihat pemesanan yang ditolak.
Hasil yang diharapkan	Sistem menampilkan halaman data pemesanan. Sistem juga dapat mengkonfirmasi data pemesanan.

Sumber : [Pengujian dan Analisis]

**Tabel 6.9 Kasus uji untuk pengujian validasi mengelola data users**

Nama Kasus Uji	Kasus Uji mengelola Data Users
Objek Uji	SRS_002_07
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas halaman untuk mengelola data users.
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem dijalankan pada keadaan sudah <i>login</i></li> <li>2. Admin memilih kolom users pada halaman utama sistem.</li> <li>3. Admin menekan tombol tambah user dan mengisi field username, password, retry password dan menekan tombol simpan.</li> </ol>
Hasil yang diharapkan	Sistem menampilkan halaman data users. Sistem juga dapat menambah data users.

Sumber : [Pengujian dan Analisis]

**Tabel 6.10 Kasus uji untuk pengujian validasi melihat info bus via SMS sah**

Nama Kasus Uji	Kasus Uji Melihat Info Bus Via SMS Sah
Objek Uji	SRS_001_01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menerima SMS dan mengolah SMS untuk dapat masuk basis data serta dapat mengirimkan SMS <i>auto-reply</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem service dijalankan.</li> <li>2. Penumpang mengirim SMS dengan format "INFO (spasi) NAMA BUS (spasi) KOTA TUJUAN (spasi) TANGGAL BERANGKAT" yang benar.</li> <li>3. Sistem dapat menerima SMS secara otomatis.</li> <li>4. Sistem mengecek ke <i>database</i>.</li> <li>5. Sistem mengirim SMS <i>auto-reply</i>.</li> </ol>
Hasil yang diharapkan	Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> info bus yang dicari penumpang.

**Tabel 6.11 Kasus uji untuk pengujian validasi melihat info bus via SMS tidak sah**

Nama Kasus Uji	Kasus Uji Melihat Info Bus Tidak Via SMS Sah
Objek Uji	SRS_001_01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat menerima SMS dan dapat mengirimkan SMS <i>auto-reply</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Aplikasi service dijalankan.</li> <li>2. Penumpang mengirimkan SMS dengan format yang salah.</li> <li>3. Sistem dapat menerima SMS secara otomatis.</li> <li>4. Sistem mengirim SMS <i>auto-reply</i>.</li> </ol>
Hasil yang diharapkan	Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.

Sumber : [Pengujian dan Analisis]

**Tabel 6.12 Kasus uji untuk pengujian validasi melihat info bus via Website**

Nama Kasus Uji	Kasus Uji Melihat Info Bus Via Website
Objek Uji	SRS_001_02
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menerima data dan mengolah data untuk dapat masuk basis data serta dapat menampilkannya di halaman <i>website</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Penumpang membuka <i>website</i></li> <li>2. Penumpang mengisi tanggal dan tujuan di pada kolom katalog tiket bus dan menekan tombol cari</li> </ol>
Hasil yang diharapkan	Sistem dapat menerima data dan mengecek ke database serta menampilkan data info bus yang dicari penumpang.

Sumber : [Pengujian dan Analisis]

**Tabel 6.13 Kasus uji untuk pengujian validasi memesan tiket via sms sah**

Nama Kasus Uji	Kasus Uji Memesan Tiket Via SMS Sah
Objek Uji	SRS_001_02
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menerima SMS dan mengolah SMS untuk dapat masuk basis data serta dapat mengirimkan SMS <i>auto-reply</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem service dijalankan.</li> <li>2. Penumpang mengirim SMS dengan format "PESAN (spasi) NAMA BUS (spasi) KOTA TUJUAN (spasi) TANGGAL BERANGKAT (spasi) JUMLAH TIKET" yang benar.</li> </ol>

	<ol style="list-style-type: none"> <li>3. Sistem dapat menerima SMS secara otomatis.</li> <li>4. Sistem mengecek ke <i>database</i>.</li> <li>5. Sistem mengirim SMS <i>auto-reply</i>.</li> </ol>
Hasil yang diharapkan	Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> konfirmasi pesanan penumpang.

Sumber : [Pengujian dan Analisis]

**Tabel 6.14 Kasus uji untuk pengujian validasi memesan tiket via sms tidak sah**

Nama Kasus Uji	Kasus Uji Memesan Tiket Via SMS Tidak Sah
Objek Uji	SRS_001_02
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat menerima SMS dan dapat mengirimkan SMS <i>auto-reply</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Aplikasi service dijalankan.</li> <li>2. Penumpang mengirimkan SMS dengan format yang salah.</li> <li>3. Sistem dapat menerima SMS secara otomatis.</li> <li>4. Sistem mengirim SMS <i>auto-reply</i>.</li> </ol>
Hasil yang diharapkan	Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.

Sumber : [Pengujian dan Analisis]

**Tabel 6.15 Kasus uji untuk pengujian validasi memesan tiket via website**

Nama Kasus Uji	Kasus Uji Memesan Tiket Via Website
Objek Uji	SRS_001_03
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menerima data masukan dan data untuk dapat masuk basis data serta dapat menampilkannya di <i>website</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Aplikasi service dijalankan</li> <li>2. Penumpang mencari tiket yang akan dipesan dikolom katalog bus</li> <li>3. Penumpang menekan tombol pesan dan mengisi field nama, alamat, email,telp/hp, jumlah tiket dan menekan tombol pesan</li> </ol>
Hasil yang diharapkan	Sistem menampilkan konfirmasi bahwa tiket berhasil dipesan.

Sumber : [Pengujian dan Analisis]

**Tabel 6.16 Kasus uji untuk pengujian validasi mengkonfirmasi pembayaran sah**

Nama Kasus Uji	Kasus Uji Mengkonfirmasi Pembayaran Sah
Objek Uji	SRS_001_04
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menerima SMS dan mengolah SMS untuk dapat masuk basis data serta dapat mengirimkan SMS <i>auto-reply</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Sistem service dijalankan.</li> <li>2. Penumpang mengirim SMS dengan format "BAYAR (spasi) NO ORDER (spasi) NO BUKTI BANK (spasi) TANGGAL KIRIM" yang benar.</li> <li>3. Sistem dapat menerima SMS secara otomatis.</li> <li>4. Admin mengecek ke <i>database</i>.</li> <li>5. Admin mengkonfirmasi data pemesanan.</li> <li>6. Sistem mengirim SMS <i>auto-reply</i>.</li> </ol>
Hasil yang diharapkan	Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> konfirmasi pesan penumpang.

Sumber : [Pengujian dan Analisis]

**Tabel 6.17 Kasus uji untuk pengujian validasi mengkonfirmasi pembayaran tidak sah**

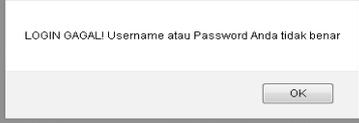
Nama Kasus Uji	Kasus Uji Mengekonfirmasi Pembayaran Tidak Sah
Objek Uji	SRS_001_04
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat menerima SMS dan dapat mengirimkan SMS <i>auto-reply</i> .
Prosedur Uji	<ol style="list-style-type: none"> <li>1. Aplikasi service dijalankan.</li> <li>2. Penumpang mengirimkan SMS dengan format yang salah.</li> <li>3. Sistem dapat menerima SMS secara otomatis.</li> <li>4. Sistem mengirim SMS <i>auto-reply</i>.</li> </ol>
Hasil yang diharapkan	Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.

Sumber : [Pengujian dan Analisis]

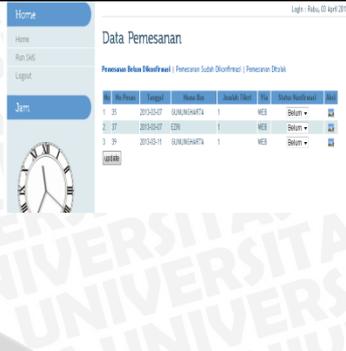
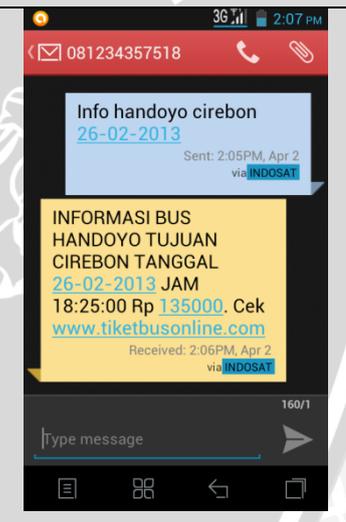
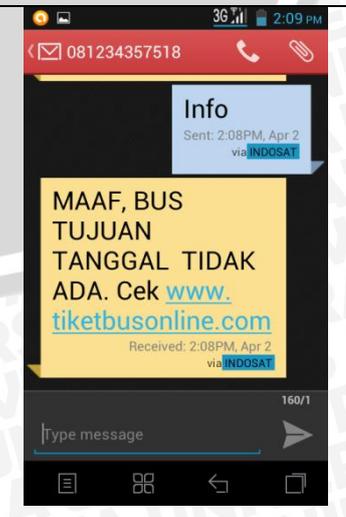
### 6.1.1.2 Hasil Pengujian Validasi

**Tabel 6.18 Hasil pengujian validasi**

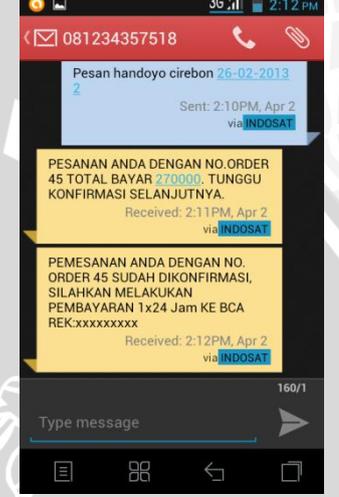
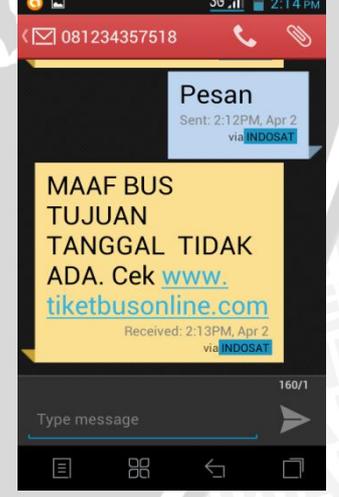
No.	Nama Kasus Uji	Hasil yang Diharapkan	Hasil yang Didapatkan	Status Validitas
-----	----------------	-----------------------	-----------------------	------------------

1	Login Sah	Sistem dapat melakukan penyeleksian kondisi <i>login</i> pada <i>database</i> berdasar data yang dimasukkan dan jika penyeleksian kondisi ini benar, maka akan masuk ke sistem sesuai dengan hak aksesnya.	Sistem dapat melakukan penyeleksian kondisi <i>login</i> pada <i>database</i> berdasar data yang dimasukkan dan jika penyeleksian kondisi ini benar, maka akan masuk ke sistem sesuai dengan hak aksesnya.	
2	Login Tidak Sah	Sistem dapat melakukan penyeleksian kondisi <i>login</i> pada <i>database</i> berdasar data yang dimasukkan dan jika penyeleksian kondisi ini salah, maka tidak akan masuk ke sistem dan aplikasi menampilkan pesan kesalahan.	Sistem dapat melakukan penyeleksian kondisi <i>login</i> pada <i>database</i> berdasar data yang dimasukkan dan jika penyeleksian kondisi ini salah, maka tidak akan masuk ke sistem dan aplikasi menampilkan pesan kesalahan.	
3	<i>Logout</i>	Sistem melakukan penghapusan session yang aktif dan menampilkan halaman <i>login</i> .	Sistem melakukan penghapusan session yang aktif dan menampilkan halaman <i>login</i> .	

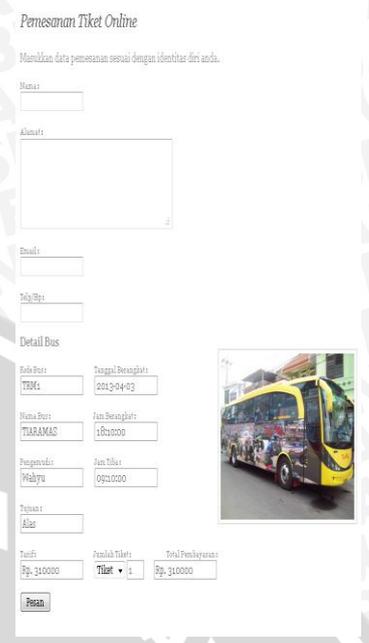
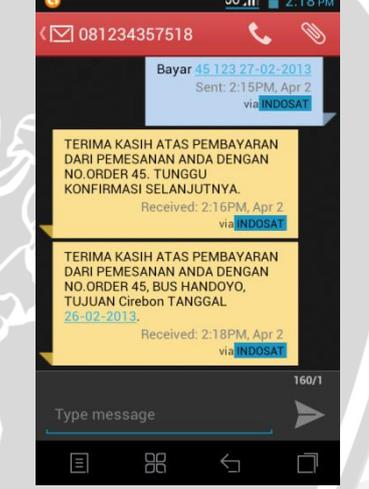
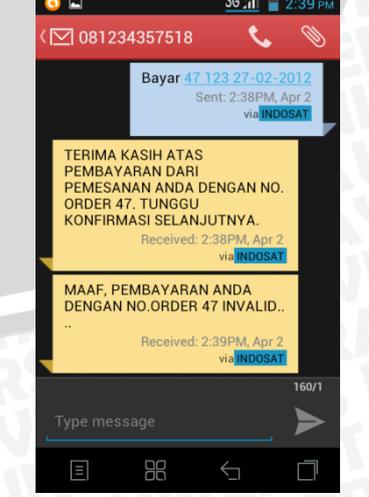
<p>4</p>	<p>Menjalankan Service SMS GATEWAY</p>	<p>Sistem menjalankan service sesuai service yang dipilih oleh admin.</p>	<p>Sistem menjalankan service sesuai service yang dipilih oleh admin.</p>	<p>Langkah-Langkah Untuk Jalankan Service</p> <p>Setting sesuai dengan hardware yang anda gunakan. Dan cobalah untuk melakukan sms terlebih dahulu setelah yang anda gunakan.</p> <p>Langkah 1. Setting GANDMOTBC                  Langkah 2. Test Kirimkan GANDMOT dengan HP                  Langkah 3. Install Database                  Langkah 4. Setting SMSDBDC                  Langkah 5. Menbuat Service GANDMOT (Black)                  Langkah 6. Menjalankan Service GANDMOT (Black)                  Langkah 7. Test Mengirim SMS (Black)                  Langkah 8. Test Menerima SMS                  Langkah 9. Menghentikan Service Gansu (Black)</p> <p>Langkah 7 - Test Mengirim SMS</p> <p>Masukkan No HP Tujuan</p> <p>Kirim Melalui Modem/HP</p> <p>Masukkan isi SMS                  (maksimum panjang SMS adalah 160 karakter)</p> <p>Kirim SMS</p>
<p>5</p>	<p>Mengelola Data Pengemudi</p>	<p>Sistem menampilkan halaman data pengemudi. Sistem juga dapat menambah, mengubah dan menghapus data pengemudi.</p>	<p>Sistem menampilkan halaman data pengemudi. Sistem juga dapat menambah, mengubah dan menghapus data pengemudi.</p>	 <p>Home</p> <p>Home</p> <p>Run SMS</p> <p>Logout</p> <p>Jam</p> <p>Id Pengemudi</p> <p>Nama Pengemudi</p> <p>Alamat</p> <p>No Telp</p> <p>Simpan Batal</p>
<p>6</p>	<p>Mengelola Data Bus</p>	<p>Sistem menampilkan halaman data bus. Sistem juga dapat menambah, mengubah, melihat dan menghapus data bus.</p>	<p>Sistem menampilkan halaman data bus. Sistem juga dapat menambah, mengubah, melihat dan menghapus data bus.</p>	 <p>Home</p> <p>Home</p> <p>Run SMS</p> <p>Logout</p> <p>Jam</p> <p>Kode Bus</p> <p>Nama Bus</p> <p>Pengemudi : Pilih Pengemudi</p> <p>Harga Bus : Rp. 0</p> <p>Gambar Bus: Browse...</p> <p>Keterangan:</p> <p>Simpan Batal</p>
<p>7</p>	<p>Mengelola Data Jadwal</p>	<p>Sistem menampilkan halaman data bus. Sistem juga dapat menambah data jadwal bus.</p>	<p>Sistem menampilkan halaman data bus. Sistem juga dapat menambah data jadwal bus.</p>	 <p>Home</p> <p>Home</p> <p>Run SMS</p> <p>Logout</p> <p>Jam</p> <p>Tujuan : Pilih Tujuan</p> <p>Tanggal Berangkat</p> <p>Jam Berangkat</p> <p>Jam Tiba</p> <p>Simpan Batal</p>

8	<p>Mengelola Data Pemesanan</p>	<p>A Sistem menampilkan halaman data pemesanan. Sistem juga dapat mengkonfirmasi data pemesanan.</p>	<p>Sistem menampilkan halaman data pemesanan. Sistem juga dapat mengkonfirmasi data pemesanan.</p>	
9	<p>Mengelola Data Users</p>	<p>Sistem menampilkan halaman data users. Sistem juga dapat menambah data users.</p>	<p>Sistem menampilkan halaman data users. Sistem juga dapat menambah data users.</p>	
10	<p>Melihat Info Bus Via SMS Sah</p>	<p>Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> info bus yang dicari penumpang.</p>	<p>Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> info bus yang dicari penumpang.</p>	
11	<p>Melihat Info Bus Via SMS Tidak Sah</p>	<p>Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.</p>	<p>Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.</p>	



12	Melihat Info Bus Via Website	Sistem dapat menerima data dan mengecek ke database serta menampilkan data info bus yang dicari penumpang	Sistem dapat menerima data dan mengecek ke database serta menampilkan data info bus yang dicari penumpang	
13	Memesan Tiket Via SMS Sah	Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> konfirmasi pesanan penumpang.	Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> konfirmasi pesanan penumpang.	
14	Memesan Tiket Via SMS Tidak Sah	Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.	Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.	



15	Memesan Tiket Via Website	Sistem menampilkan konfirmasi bahwa tiket berhasil dipesan.	Sistem menampilkan konfirmasi bahwa tiket berhasil dipesan.	
16	Mengkonfirmasi Pembayaran Sah	Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> konfirmasi pesanan penumpang.	Sistem dapat menerima SMS dan mengecek ke database serta mengirim SMS <i>auto-reply</i> konfirmasi pesanan penumpang.	
17	Mengkonfirmasi Pembayaran Tidak Sah	Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.	Sistem dapat menerima SMS dan mengirim SMS <i>auto-reply</i> berupa kesalahan format SMS.	

Sumber : [Pengujian dan Analisis]

### 6.1.2 Pengujian Performa

Pengujian performa digunakan untuk mengetahui apakah sistem yang dibangun dapat memberikan *auto-reply* SMS dengan cepat, sehingga dapat memberikan kepastian kepada penumpang dengan cepat pula. Pengujian ini dilakukan dengan menghitung rata-rata lama waktu antara sms yang masuk dengan sms balasan. Pada skripsi ini dilakukan pengujian performa terhadap perangkat lunak SMS Gateway pada Sistem Reservasi Tiket di Terminal Arjosari Malang.

Tabel 6.19 Hasil pengujian performa

<i>id</i> SMS	Nomor Pengirim	Waktu Terima (t)	Waktu Kirim (k)	Waktu Tunggu (k – t) (seconds)
1	+6281331166073	08:45:32	08:46:02	30
2	+6285394559972	09:50:12	09:50:47	35
3	+6285755676620	09:55:11	09:55:58	47
4	+6285646477963	12:12:55	12:13:47	52
5	+628970418907	12:30:22	12:30:58	36
6	+6285646412251	12:33:15	12:33:42	27
7	+6285797284663	12:45:10	12:46:05	55
8	+6285646596965	12:52:03	12:52:41	38
9	+6285655505963	12:59:01	12:59:33	32
10	+6285646496611	13:14:45	13:15:10	25
11	+6285646584080	13:26:22	13:27:07	45
12	+6285655541353	13:30:21	13:30:57	36
13	+6281805063617	07:34:24	07:35:71	47
14	+6287819544335	07:48:54	07:49:46	52
15	+6282331157795	07:59:02	07:59:48	46
16	+6283834745261	08:22:01	08:22:41	40
17	+6282143685055	08:35:20	08:35:57	37
18	+6281336516651	08:55:14	08:56:01	47
19	+628170500090	09:01:34	09:02:25	51
20	+6283848557985	09:12:29	09:13:03	34
21	+623417000849	09:30:29	09:31:25	46

22	+6281808000751	09:53:39	09:54:24	45
23	+6283848557983	10:01:24	10:02:09	45
24	+6283848557981	11:01:22	11:01:57	35
25	+6283848557983	11:29:35	11:30:40	65
<b>Jumlah Rata-Rata Waktu Tunggu</b>				<b>41.92</b>

Sumber : [Pengujian dan Analisis]

Dari tabel 6.18 dapat disimpulkan bahwa waktu tunggu antara SMS masuk dan dibalas oleh aplikasi adalah tidak sama. Dengan melakukan 25 pengujian, maka diperoleh rata-rata waktu tunggu *auto-replySMS* dalam aplikasi perangkat lunak *SMS Gateway* pada Sistem Reservasi Tiket di Terminal Arjosari Malang adalah 41.92detik.

### 6.1.3 Pengujian Usability

Pengujian *usability* digunakan untuk menguji perangkat lunak Sistem Reservasi Tiket di Terminal Arjosari Malang. Pengujian ini ditujukan untuk mengetahui seberapa besar perangkat lunak Sistem Reservasi Tiket Bus di Terminal Arjosari Malang bisa digunakan dengan baik oleh para calon *user*. Pengujian ini dilakukan dengan cara menyebarkan kuisioner kepada para calon *user* dan beberapa pegawai P.O bus di Terminal Arjosari Malang. Calon *user* pada pengujian ini adalah calon penumpang bus di Terminal Arjosari Malang. Pada pengujian ini kriteria – kriteria khusus pada sampel seperti rentang usia, jenis kelamin, dan latar belakang pendidikan diabaikan. Jumlah sampel calon *user* yang diuji sebanyak 22 sampel. Komponen, pertanyaan yang digunakan pada pengujian ini beserta hasil kuisioner ditunjukkan pada Tabel 6.20.

**Tabel 6.20** Komponen dan pertanyaan pengujian *usability*

No	Komponen	Pertanyaan	Jawaban	Jawaban	Jawaban
			“Cukup”	“Kurang”	“Tidak”
1	Isi Website	apakah website ini menyediakan informasi yang tepat yang anda butuhkan ?	18	4	0
		apakah kandungan	14	7	1

		informasi memenuhi kebutuhan anda ?			
		apakah website ini menyediakan laporan hanya tentang yang anda butuhkan ?	9	11	2
		apakah website ini menyediakan informasi yang memadai ?	17	5	0
2	Ketepatan Informasi	apakah informasi yang diberikan sistem akurat ?	16	6	0
		apakah anda puas dengan keakuratan informasi yang diberikan ?	18	4	0
3	Struktur Format dan Desain Sistem	apakah output yang disajikan dalam format yang berguna ?	17	5	0
		apakah informasi yang diberikan jelas ?	12	9	1
		apakah situs web menyediakan desain yang baik ?	13	9	0
		kemudahan penggunaan	18	4	0
		apakah website ini user friendly (intuitif digunakan) ?	12	10	0
4	Ketanggapan Informasi	apakah anda mendapatkan informasi yang anda butuhkan sewaktu-waktu ?	16	6	0
		apakah website ini menyediakan informasi terkini ?	12	9	1
5	Sikap Penggunaan	apakah menggunakan website ini adalah ide	20	2	0

	Sistem	yang baik ?			
		apakah menggunakan website ini menyenangkan ?	16	5	1
		apakah menggunakan website ini bermanfaat untuk anda ?	19	3	0
6	Niat Penggunaan Sistem	apakah anda berniat menggunakan website ini dalam keseharian ?	17	3	2
		apakah anda berniat untuk menjadi pengguna setia website ini ?	6	14	2

Sumber : [Pengujian]

Dari pertanyaan di atas didapatkan 20 sampel dengan total 360 jawaban pengujian. Total dari hasil pengujian usability ini ditunjukkan pada Tabel 6.21.

**Tabel 6.21 Hasil pengujian usability**

No	Komponen	Jawaban "Cukup"	Jawaban "Kurang"	Jawaban "Tidak"
1	Isi Website	58	27	3
2	Ketepatan Informasi	34	10	0
3	Struktur Format dan Desain Sistem	72	36	1
4	Ketanggapan Informasi	28	15	1
5	Sikap Penggunaan Sistem	55	10	1
6	Niat Penggunaan Sistem	23	17	4
	<b>Total</b>	270	116	10

Sumber : [Pengujian]

Dari hasil pengujian di atas dapat didapatkan 270 jawaban "cukup", 116 jawaban "Kurang" dan 10 Jawaban "Tidak" dengan total semua jawaban 360. Dapat disimpulkan bahwa nilai persentase jawaban "Cukup" sebesar 75%. Rincian nilai persentase tiap komponen dijelaskan pada Tabel 6.22.

**Tabel 6.22 Rincian nilai persentase tiap komponen**

No	Komponen	Persentase
1	Isi Website	65%
2	Ketepatan Informasi	77%
3	Struktur Format dan Desain Sistem	66%
4	Ketanggapan Informasi	63%

5	Sikap Penggunaan Sistem	83%
6	Niat Penggunaan Sistem	52%

Sumber : [Pengujian]

Nilai persentase tertinggi dari komponen di atas adalah sebesar 83% pada komponen Sikap Penggunaan Sistem, dan nilai terendah sebesar 52% pada komponen Niat Penggunaan Sistem.

## 6.2 ANALISIS

Proses analisis bertujuan untuk mendapatkan kesimpulan dari hasil pengujian Sistem Reservasi Tiket di Terminal Arjosari 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, analisis hasil pengujian performa dan analisis hasil pengujian Usability.

### 6.2.1 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 SMS *Gateway* pada Sistem Reservasi Tiket di Terminal Arjosari telah memenuhi kebutuhan yang telah dijabarkan pada tahap analisis kebutuhan.

### 6.2.2 Analisis Hasil Pengujian Performa

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

- a. Provider dan tempat berpengaruh terhadap lama waktu tunggu *auto-reply* SMS.

- b. Dalam pengujian perangkat lunak SMS *Gateway* pada Sistem Reservasi Tiket di Terminal Arjosari yang dilakukan pada *notebook* Sony Vaio VGN CR-353, didapat hasil lama waktu tunggu *auto-reply* yang rata – rata sebesar 41.92 detik.

### 6.2.3 Analisis Hasil Pengujian Usability

Proses analisis terhadap hasil pengujian dilakukan dengan melihat nilai persentase jawaban “Cukup”. Dari 6 komponen pengujian, ada 2 komponen dengan nilai dibawah 65%, yaitu Ketanggapan Informasi dan Niat Penggunaan Sistem dengan nilai masing-masing 63% dan 52%.

Dari data kuesioner, pada komponen Ketanggapan Informasi total jawaban “Cukup” sebanyak 28, jawaban “Kurang” sebanyak 15 dan jawaban “Tidak” sebanyak 1. Dari dua pertanyaan yang diajukan, pertanyaan kedua yaitu “Apakah website ini menyediakan informasi terkini ?” memiliki perolehan jawaban “Cukup” sebanyak 12 jawaban, jawaban “Kurang” sebanyak 9 jawaban dan jawaban “Tidak” sebanyak 1 jawaban. Calon user menginginkan informasi yang lebih detail, terkini dan mudah dimengerti tentang SMS konfirmasi, data-data bus seperti nama bus, jadwal, tujuan dan nomor tempat duduk. . Inilah yang menyebabkan ada 15 jawaban “Kurang” pada komponen Ketanggapan Informasi.

Dari data kuesioner, pada komponen Niat Penggunaan Sistem total jawaban “Cukup” sebanyak 21, jawaban “Kurang” sebanyak 15 dan jawaban “Tidak” sebanyak 4. Dari dua pertanyaan yang diajukan, pertanyaan kedua yaitu “apakah anda berniat untuk menjadi pengguna setia website ini ?” memiliki perolehan jawaban “Cukup” sebanyak 6 jawaban, jawaban “Kurang” sebanyak 12 jawaban dan jawaban “Tidak” sebanyak 2 jawaban. Beberapa calon user masih terbiasa dengan pemesanan tiket secara manual jadi masih terlihat canggung dengan pemesanan tiket secara online maupun via SMS. Inilah yang menyebabkan ada 15 jawaban “Kurang” pada komponen Niat Penggunaan Sistem. Dari total keseluruhan persentase jawaban, didapatkan nilai 68%.

## BAB VII PENUTUP

### 7.1 KESIMPULAN

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

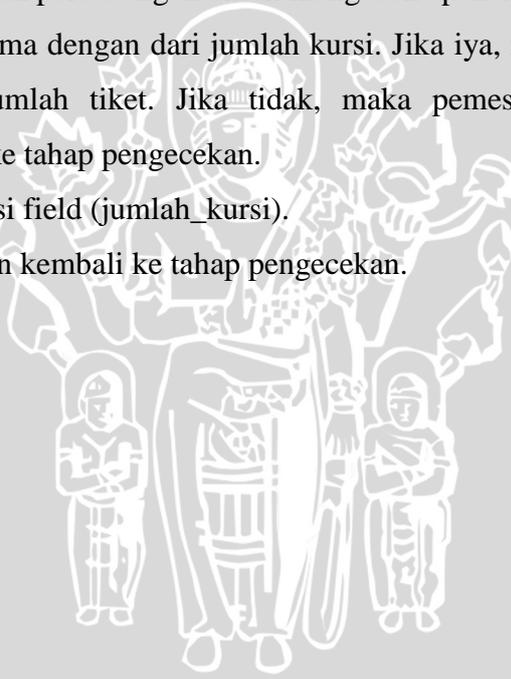
1. Aplikasi Penumpang perangkat lunak pada sistem reservasi tiket di Terminal Arjosari Malang dapat digunakan untuk melihat info bus, memesan tiket bus, dan mengkonfirmasi pembayaran tiket.
2. Aplikasi Admin perangkat lunak pada sistem reservasi tiket di Terminal Arjosari Malang dapat digunakan untuk mengelola data bus dan reservasi tiket.
3. Dalam pengujian perangkat lunak SMS Gateway pada Sistem Reservasi Tiket di Terminal Arjosari yang dilakukan pada *notebook* Sony Vaio VGN CR-353, didapat hasil lama waktu tunggu *auto-reply* yang rata – rata sebesar 41.92 detik. Provider, posisi dan banyaknya jumlah SMS yang masuk ke sistem berpengaruh terhadap lama waktu tunggu *auto-reply* SMS.
4. Dalam persentase jawaban pengujian *usability*, didapatkan nilai 67,7%.

### 7.2 SARAN

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

1. Disarankan memilih *provider* untuk *server* yang memiliki *traffic* kecil.
2. Dapat ditambahkan beberapa fitur seperti pemilihan nomor tempat duduk dan pedoman pemesanan tiket bagi user dalam sistem reservasi tiket bus.
3. Disarankan untuk memberikan keamanan *backup* data antrian untuk setiap harinya.
4. Disarankan untuk memperbaiki desain dan tampilan web agar lebih menarik minat user.

5. Disarankan untuk menyediakan informasi yang lebih detail, terkini dan mudah dimengerti tentang SMS konfirmasi, data-data bus seperti nama bus, jadwal, tujuan dan nomor tempat duduk.
6. Untuk pengembangan lebih lanjut dapat ditambahkan perancangan untuk pembatasan jumlah tiket pada sistem reservasi secara otomatis.
7. Disarankan untuk menambahkan skenario pengujian performa untuk banyak SMS.
8. Untuk kedepannya disarankan untuk mengubah fungsi kerja admin dalam mengkonfirmasi pemesanan dari manual menjadi otomatis dengan cara sebagai berikut :
  - a) Menambahkan 1 field (jumlah\_kursi) pada tabel bus.
  - b) Melakukan percabangan untuk mengecek apakah jumlah tiket lebih kecil atau sama dengan dari jumlah kursi. Jika iya, maka jumlah kursi dikurangi jumlah tiket. Jika tidak, maka pemesanan ditolak lalu kembali ke ke tahap pengecekan.
  - c) Update isi field (jumlah\_kursi).
  - d) Pengujian kembali ke tahap pengecekan.



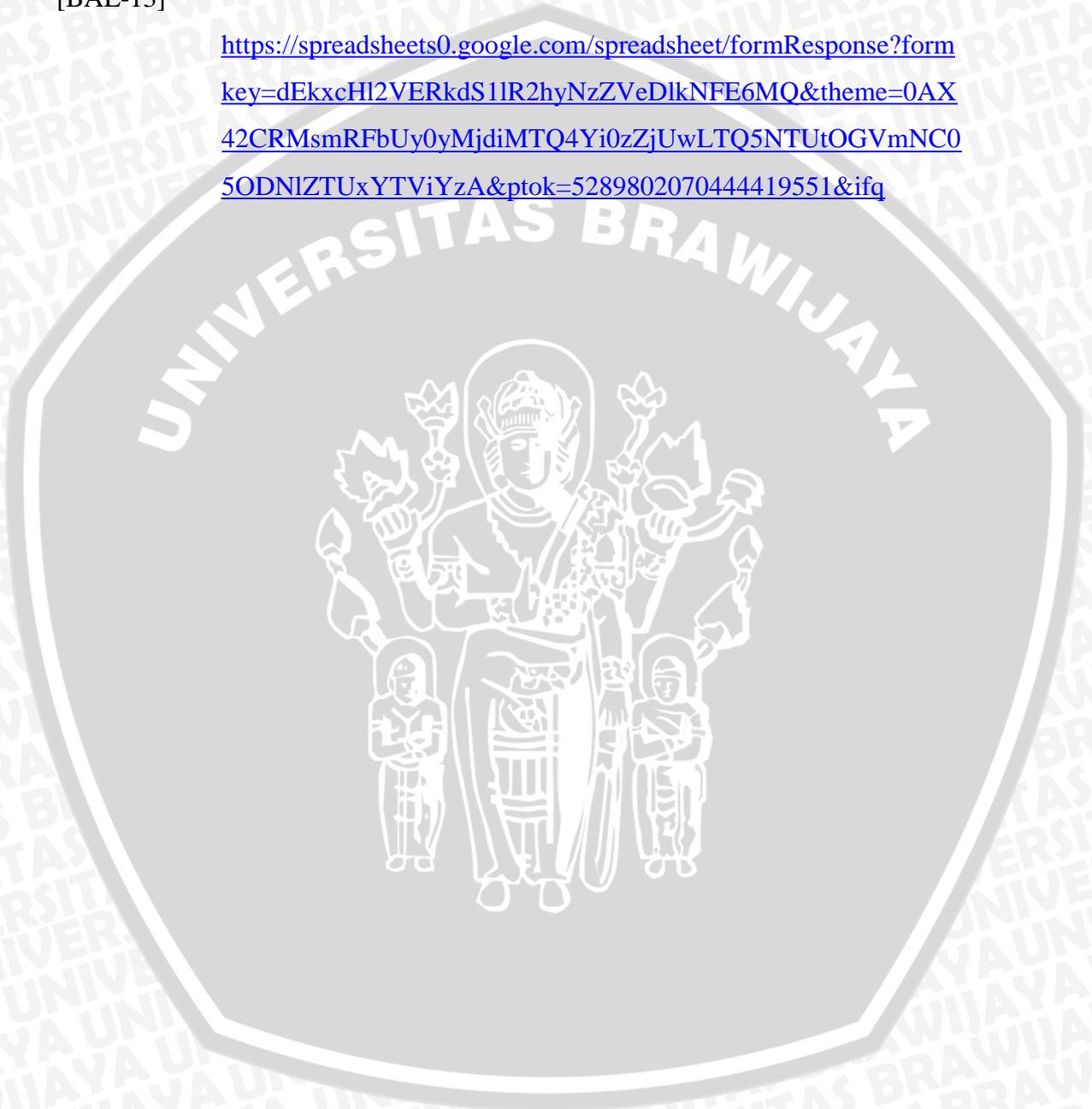
## DAFTAR PUSTAKA

- [BOO-05] Booch, Grady, James Rumbaugh, Ivar Jacobson. 2005. *The Unified Modelling Language User Guide: Second Edition*. Addison Wesley.
- [DAV-09] David. 2009. Mengenal Cara Kerja SMS Gateway. Wordpress. Online.  
<http://duniadhana.wordpress.com/2009/08/02/mengenal-cara-kerja-sms-gateway/>. 23 September 2011.
- [JAK-03] Nielsen, Jakob. 2003. *Usability 101 :Introduction to Usability*. Akses dari <http://faculty.wiu.edu/CB-Dilger/f07/480/readings/10-nielsen-usability-101.pdf>.
- [MAS-11] Murtadlo, Ali & Arifin, Firman & Setiawardhana. 2011. Simulasi Sistem Informasi Posisi Kereta Api dengan Menggunakan GPS untuk Keselamatan Penumpang. Surabaya : Institut Teknologi Sepuluh November.
- [MAZ-10] Mazarie. 2010. Mengenal Bagaimana Cara Kerja SMS Web Online. <http://mazarie.web.id/mengenal-bagaimana-cara-kerja-sms-short-message-service.html>. 23 September 2011
- [PAN-09] Pandana, Harja. 2009. Perancangan Sistem Informasi Persediaan Metode Entity Relationship Diagram dan Bahasa pemrograman PHP di PT. Tiga Musim Mas Jaya. Program Pendidikan Sarjana Ekstensi Departemen Teknik Industri Fakultas Teknik Universitas Sumatera Utara Medan.
- [PRE-10] Pressman, Roger S. 2010. *Software Engineering : A Practitioner's Approach, Fifth Edition*. McGraw Hill.
- [RAM-12] Ramadhika A. 2012. SMS Gateway menggunakan Gammu dan MySQL. [http://www.ubaya.ac.id/ubaya/articles\\_detail/33/SMS-Gateway-menggunakan-Gammu-dan-MySQL.html](http://www.ubaya.ac.id/ubaya/articles_detail/33/SMS-Gateway-menggunakan-Gammu-dan-MySQL.html).
- [ROS-11] A. S., Rosa & Shalahudin, M. 2011. *Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung: Modula.

[SID-04] Sidik, Betha, Ir. 2004. Pemrograman Web dengan PHP, Cetakan Pertama. Bandung : Penerbit Informatika.

[BAL-13]

<https://spreadsheets0.google.com/spreadsheet/formResponse?formkey=dEkxcHI2VERkdS1IR2hyNzZVeDIkNFE6MQ&theme=0AX42CRMsmRFbUy0yMjdiMTQ4Yi0zZjUwLTQ5NTUtOGVmNC05ODNIZTUxYTViYzA&ptok=5289802070444419551&ifq>



## Lampiran 1

### Kuesioner Website Sistem Reservasi Tiket di Terminal Arjosari Malang

P.O :

#### **Berikan tanggapan anda terhadap isi dari website ini**

1. apakah website ini menyediakan informasi yang tepat yang anda butuhkan  
a. cukup      b. kurang      c. tidak
2. apakah kandungan informasi memenuhi kebutuhan anda  
a. cukup      b. kurang      c. tidak
3. apakah website ini menyediakan laporan hanya tentang yang anda butuhkan  
a. cukup      b. kurang      c. tidak
4. apakah website ini menyediakan informasi yang memadai  
a. cukup      b. kurang      c. tidak

#### **Berikan tanggapan anda terhadap ketepatan informasi**

1. apakah informasi yang diberikan sistem akurat  
a. cukup      b. kurang      c. tidak
2. apakah anda puas dengan keakuratan informasi yang diberikan  
a. cukup      b. kurang      c. tidak

#### **Berikan tanggapan anda terhadap struktur format serta desain dari website ini**

1. apakah output yang disajikan dalam format yang berguna  
a. cukup      b. kurang      c. tidak
2. apakah informasi yang diberikan jelas  
a. cukup      b. kurang      c. tidak
3. apakah situs web menyediakan desain yang baik  
a. cukup      b. kurang      c. tidak
4. kemudahan penggunaan  
a. cukup      b. kurang      c. tidak
5. apakah website ini user friendly (intuitif digunakan)  
a. cukup      b. kurang      c. tidak

#### **Berikan tanggapan anda terhadap ketanggapan informasi**

1. apakah anda mendapatkan informasi yang anda butuhkan sewaktu-waktu  
a. cukup      b. kurang      c. tidak
2. apakah website ini menyediakan informasi terkini  
a. cukup      b. kurang      c. tidak

#### **Sikap anda dalam menggunakan sistem**

1. apakah menggunakan website ini adalah ide yang baik  
a. cukup      b. kurang      c. tidak
2. apakah menggunakan website ini menyenangkan  
a. cukup      b. kurang      c. tidak
3. apakah menggunakan website ini bermanfaat untuk anda  
a. cukup      b. kurang      c. tidak

#### **Niat untuk menggunakan sistem ini**

1. apakah anda berniat menggunakan website ini dalam keseharian  
a. cukup      b. kurang      c. tidak
2. apakah anda berniat untuk menjadi pengguna setia website ini  
a. cukup      b. kurang      c. tidak

Kritik dan Saran :

