

**PERANCANGAN APLIKASI UNTUK *CUSTOMER SERVICE*
BERBASIS *WEB* MENGGUNAKAN *SMS GATEWAY*
(Studi Kasus PT. Jaya Angkasa Globalindo)**

TUGAS AKHIR

Oleh:

ERLANGGA PRASETIO
361401022



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER
INDONESIA MANDIRI
BANDUNG
2018**

LEMBAR PENGESAHAN

**PERANCANGAN APLIKASI UNTUK *CUSTOMER SERVICE*
BERBASIS *WEB* MENGGUNAKAN *SMS GATEWAY*
(Studi Kasus PT. Jaya Angkasa Globalindo)**

***DESIGNING APPLICATIONS FOR CUSTOMER SERVICE
WEB BASED WITH USING THE SMS GATEWAY
(CASE STUDY PT. Jaya Angkasa Globalindo)***

Oleh:

**ERLANGGA PRASETIO
NIM. 361401022**

Tugas Akhir ini telah diterima dan disahkan
Untuk memenuhi persyaratan mencapai gelar
Sarjana Teknik Informatika

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER
INDONESIA MANDIRI**

Bandung, 21 Oktober 2018

Disahkan oleh

Ketua Program Studi,

Dosen Pembimbing

Chalifa Chazar, S.T., M.T.
NIDN : 0421098704

Patah Herwanto, ST.,M.Kom
NIDN : 0027107501

LEMBAR PERSETUJUAN REVISI TUGAS AKHIR

**PERANCANGAN APLIKASI UNTUK *CUSTOMER SERVICE*
BERBASIS *WEB* MENGGUNAKAN *SMS GATEWAY*
(Studi Kasus PT. Jaya Angkasa Globalindo)**

***DESIGNING APPLICATIONS FOR CUSTOMER SERVICE
WEB BASED WITH USING THE SMS GATEWAY
(CASE STUDY PT. Jaya Angkasa Globalindo)***

Telah melakukan sidang Tugas Akhir pada hari Rabu, 3 Oktober 2018 dan telah melakukan revisi sesuai dengan masukan pada saat sidang Tugas Akhir

Menyetujui,

No	Nama	Keterangan	Tanda Tangan
1	Patah Herwanto, ST.,M.Kom.	Pembimbing	
2	Dr. Chairuddin, M.T.,M.M.	Penguji 1	
3	Dede Suryana, Ir., M.Si	Penguji 2	

Bandung, 21 Oktober 2018
Mengetahui
Ketua Program Studi

Chalifa Chazar, S.T., M.T.
NIDN : 0421098704

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa:

1. Tugas akhir ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri maupun perguruan Tinggi lainnya.
2. tugas akhir ini murni merupakan karya penelitian saya sendiri dan tidak menjiplak karya pihak lain. Dalam hal ada bantuan atau arahan dari pihak lain maka telah saya sebutkan identitas dan jenis bantuannya di dalam lembar ucapan terima kasih.
3. Seandainya ada karya pihak lain yang ternyata memiliki kemiripan dengan karya saya ini, maka hal ini adalah di luar pengetahuan saya dan terjadi tanpa kesengajaan dari pihak saya.

Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terbukti adanya kebohongan dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai norma yang berlaku di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.

Bandung, 21 Oktober 2018

Yang Membuat Pernyataan

(Erlangga Prasetio)

361401022

ABSTRAK

Era teknologi informasi ditandai dengan kemudahan dan kecepatan aliran informasi dari satu komunitas ke yang lainnya. Salah satu produk teknologi informasi adalah telekomunikasi menggunakan perangkat *Handphone* atau telepon seluler. Di dalamnya terdapat sebuah metode komunikasi yang dinamakan SMS (*Short Message Service*) yaitu sebuah konsep pengiriman informasi berbasis SMS.

SMS (*Short Message Service*) merupakan salah satu komunikasi dengan teks melalui perangkat bergerak (*mobile device*). Perkembangan telekomunikasi yang begitu pesat seiring dengan perkembangan teknologi informasi memungkinkan pengaksesan data lebih praktis, lebih cepat dan efisien. SMS *Gateway* merupakan alat yang digunakan untuk mengirim dan menerima informasi kepada *customer* dan menggunakan *web* untuk memberikan tampilan antarmuka yang lebih mudah dipahami dan di mengerti.

Metode yang digunakan dalam membangun aplikasi ini adalah metode pengembangan perangkat lunak model *waterfall*. Aplikasi ini dibangun untuk mendukung dan membantu *customer service* dalam memberikan informasi kepada *customer* secara perorangan atau secara tersiar dan juga pesan terjadwal dalam memberikan informasi, dikarenakan didalam perusahaan masuk kurang cukup baik dalam memberikan informasi kepada *customer* masih secara manual, diharapkan aplikasi ini dapat membantu kinerja *customer service* dalam memberikan informasi secara efektif dan efisien sebagai pengganti pekerjaan manual.

Kata kunci: SMS (*Short Message Service*), SMS Gateway

ABSTRACT

The era of information technology is characterized by ease and speed the flow of information from one community to another. One of the products of information technology is the Mobile device using telecommunications or cellular phone. In it there is a communication method called SMS (Short Message Service) which is an SMS-based information delivery concept.

SMS (Short Message Service) is one of the communication with the text through mobile device (mobile device). Rapid telecommunications development in line with the information technology developments allow data access easier, faster and more efficient. SMS Gateway is a tool that is used to send and receive information to the customer and use the web to give you the look of an interface that is more easily understood and in understanding.

The methods used in developing this application is a method of waterfall model of software development. These applications are built to support and assist customer service in providing information to the customer individually or in Word and also scheduled messages in providing information, due to the company's entry in the less-good enough in providing information to the customer are still manually, hopefully this application can help the performance of customer service in providing information effectively and efficiently in lieu of manual work.

Key word: SMS (Short Message Service), SMS Gateway

UCAPAN TERIMA KASIH

Tugas Akhir ini disusun untuk melengkapi program perkuliahan Strata Satu Jurusan Jurusan Teknik Informatika. Dalam penyusunan laporan ini, tidak lepas dari bantuan berbagai pihak, baik secara moril maupun materil walaupun hasilnya masih jauh dari kata sempurna. Dengan Penuh rasa syukur kepada Allah swt, penulis ingin menyampaikan rasa terima kasih kepada :

1. Patah Herwanto. ST., M.Kom selaku dosen pembimbing, terimakasih atas waktu, bimbingan, dan arahan yang telah diberikan kepada penulis dalam menyelesaikan Tugas Akhir ini.
2. Bapak Dr. Chairuddin, Ir., M.M., M.T selaku Ketua Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.
3. Chalifa Chazar, S. T, M.T selaku ketua program studi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.
4. Seluruh dosen pengajar STMIK-IM yang telah memberikan bekal ilmu pengetahuan kepada penulis.
5. Kedua Orang tuaku tercinta, adik, dan saudara-saudaraku yang telah memberikan dukungan begitu banyak baik moril maupun materil serta nasehat, semangat kepercayaan dan do'a yang tiada henti.
6. Kepada teman-teman seperjuangan STMIK STIE STAN – Indonesia Mandiri yang tiada henti menemani, mendo'akan dan mendukung penulis.
7. Tri Hardianti sang kekasih, motivator pribadi, sang calon pendamping wisuda yang tanpa henti selalu memberikan dukungan dan semangat. Nasihat dan saran yang ia berikan adalah hal yang

menolong dan membuat saya tersadar untuk berusaha lebih baik dan bekerja lebih keras dari sebelumnya. Kalimat penenang yang ia berikan adalah hal yang membuat saya dapat bangkit dan tidak takut lagi ketika berbagai tamparan dan teguran keras saya peroleh dan membuat saya merasa putus asa. *Thank you for being who you are and for being with me.*

8. Kepada teman-teman seperjuangan Teknik Informatika angkatan 2014, yang sama sama berjuang dalam menyusun tugas akhir, yang saling membantu dan memberikan semangat dalam menyelesaikan Tugas Akhir ini.
9. Seluruh rekan, sahabat, dan pihak-pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu penulis baik secara langsung maupun tidak langsung yang memberikan semangat kepada penulis dalam menyelesaikan Tugas Akhir ini.

Dalam melakukan penyusunan Tugas Akhir ini, penulis juga menghadapi kesulitan. Penulis menyadari bahwa dalam penyusunan Tugas Akhir ini masih banyak kekurangan dan kesalahan. Oleh karena itu penulis mohon maaf atas adanya kekurangan dan penulis mengharapkan adanya saran yang dapat membangun. Akhir kata, Penulis berharap semoga Laporan Tugas Akhir ini bermanfaat bagi penulis dan para pembaca.

Bandung, 21 Oktober 2018

penulis

(Erlangga Prasetio)

361401022

KATA PENGANTAR

Puji dan syukur atas kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul “**PERANCANGAN APLIKASI UNTUK *CUSTOMER SERVICE* BERBASIS *WEB* MENGGUNAKAN *SMS GATEWAY* (Studi Kasus PT. Jaya Angkasa Globalindo)**”. Tugas Akhir ini disusun sebagai salah satu syarat dalam penyelesaian program pendidikan strata satu (S1) Jurusan Teknik Informatika pada Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.

Dengan segala kerendahan hati penulis menyadari bahwa Tugas Akhir ini masih jauh dari kesempurnaan karena keterbatasan kemampuan penulis. Namun demikian diharapkan agar Tugas Akhir ini dapat memenuhi syarat yang diperlukan.

Bandung, 21 Oktober 2018

Penulis

(Erlangga Prasetio)
361401022

DAFTAR ISI

LEMBAR PENGESAHAN	
LEMBAR PERSETUJUAN REVISI TUGAS AKHIR	
SURAT PERNYATAAN	i
ABSTRAK	ii
ABSTRACK	iii
UCAPAN TERIMAKASIH	iv
KATA PENGHANTAR	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiv
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Identifikasi Masalah	2
1.3. Tujuan Penelitian.....	3
1.4. Batasan Masalah	3
1.5. Metode Penelitian	4
1.5.1. Teknik Pengumpulan Data.....	4
1.5.2. Metode Perancangan.....	4
1.6. Sistematika Penulisan	7
BAB II LANDASAN TEORI	9
2.1. Pengertian Aplikasi.....	9
2.2. Definisi Sistem	10
2.2.1. Karakteristik Sistem	11
2.3. <i>Customer Service</i>	15

2.4.	<i>Short Massaging Services (SMS)</i>	16
2.4.1.	Karkteristik SMS	16
2.4.2.	<i>SMS Center (SMSC)</i>	17
2.4.3.	Mangfaat Pengguna SMS	17
2.5.	Pengertian <i>SMS Gateway</i>	18
2.6.	<i>Gammu</i>	21
2.7.	Unsur-Unsur <i>Website</i>	22
2.7.1.	HTML.....	22
2.7.2.	<i>Web Browser</i>	23
2.7.3.	<i>Web Server</i>	23
2.7.4.	PHP.....	23
2.7.5.	SQL.....	24
2.7.6.	<i>Cascade Sheet Style (CSS)</i>	24
2.7.7.	<i>CodeIgniter</i>	24
2.8.	Metode <i>Waterfall</i>	25
2.9.	Diagram Alir Dokumen (<i>flowMap</i>)	27
2.10.	<i>Unified Modeling Language (UML)</i>	28
2.10.1.	<i>Use Case Diagram</i>	29
2.10.2.	<i>Activity Diagram</i>	30
2.10.3.	<i>Class Diagram</i>	31
2.10.4.	<i>Sequence Diagram</i>	32
2.11.	Basis Data	34
2.11.1.	MySQL	34
2.11.2.	PHPMYAdmin	35
2.12.	<i>Entity Relationship Diagram (ERD)</i>	35
2.13.	XAMPP	37
2.14.	<i>Black Box Testing</i>	38

BAB III PEMBAHASAN	39
3.1. Analisis	39
3.1.1. Analisis Perusahaan	39
3.1.1.1. Visi PT. Jaya Angkasa Globalindo	41
3.1.1.2. Misi PT. Jaya Angkasa Globalindo.....	41
3.1.1.3. Leglitas Perusahaan.....	42
3.1.1.4. Struktur Organisasi.....	43
3.1.2. Analisis Sistem yang Sedang Berjalan	43
3.1.3. Analisis Permasalahan	46
3.1.4. Analisis Sistem yang Diusulkan	46
3.1.5. Analisis Perangkat yang digunakan.....	47
3.1.5.1. Perangkat Keras (<i>Hardware</i>)	47
3.1.5.2. Perangkat Lunak (<i>Software</i>).....	48
3.2. Desain	48
3.2.1. Perancangan Sistem.....	48
3.2.1.1. <i>Use Case Diagram</i>	49
3.2.1.2. <i>Activity Diagram</i>	65
3.2.1.3. <i>Sequence Diagram</i>	90
3.3.1.4. <i>Class Diagram</i>	98
3.2.1.5. <i>Entity Relationship Diagram (ERD)</i>	100
3.2.1.6. <i>Design Interface</i>	105
3.3. Pengodean	111
3.3.1. Implementasi	111
3.4. Pengujian.....	119
3.4.1. Rencana Pengujian	120
3.4.2. Kasus dan Hasil Pengujian	120
BAB IV PENUTUP	127

4.1. Kesimpulan	127
4.2. Saran	128
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR GAMBAR

GAMBAR 1.1. Ilustrasi Model <i>Waterfall</i> (Rosa A.S dan M.Shalahuddin, 2018)..5	5
GAMBAR 2.1. Karakteristik sistem (Tata Sutabri, 2012)	14
GAMBAR 2.2. Alur kerja SMS <i>Gateway</i> (Ardana, 2004:35)	21
GAMBAR 2.3. Ilustrasi Model <i>Waterfall</i> (Rosa A.S dan M.Shalahuddin, 2018)25	25
GAMBAR 2.4. <i>Control Panel</i> Xampp	37
GAMBAR 3.1. Struktur Organisasi PT. Jaya Angkasa Globalindo	43
GAMBAR 3.2. Alur <i>Flowmap</i> yang sedang berjalan	45
GAMBAR 3.3. <i>Use Case Diagram</i> aplikasi untuk <i>customer service</i>	52
GAMBAR 3.4. <i>Activity Diagram Login</i>	66
GAMBAR 3.5. <i>Activity Diagram</i> pengiriman pesan	67
GAMBAR 3.6. <i>Activity Diagram</i> pengiriman pesan tersiar	68
GAMBAR 3.7. <i>Activity Diagram</i> Mengelola pesan terjadwal.....	69
GAMBAR 3.8. <i>Activity Diagram</i> tambah pesan terjadwal	70
GAMBAR 3.9. <i>Activity Diagram</i> edit pesan terjadwal.....	71
GAMBAR 3.10. <i>Activity Diagram</i> hapus pesan terjadwal	72
GAMBAR 3.11. <i>Activity Diagram</i> Mengelola kontak.....	73
GAMBAR 3.12. <i>Activity Diagram</i> tambah kontak	74
GAMBAR 3.13. <i>Activity Diagram</i> cari kontak.....	75
GAMBAR 3.14. <i>Activity Diagram</i> edit kontak.....	76
GAMBAR 3.15. <i>Activity Diagram</i> hapus kontak.....	77
GAMBAR 3.16. <i>Activity Diagram</i> Mengelola kontak grup	78
GAMBAR 3.17. <i>Activity Diagram</i> tambah grup.....	79
GAMBAR 3.18. <i>Activity Diagram</i> cari grup	80
GAMBAR 3.19. <i>Activity Diagram</i> edit grup	81

GAMBAR 3.20. <i>Activity Diagram</i> hapus grup	82
GAMBAR 3.21. <i>Activity Diagram</i> Mengelola kontak masuk	83
GAMBAR 3.22. <i>Activity Diagram</i> balas kontak masuk	84
GAMBAR 3.23. <i>Activity Diagram</i> hapus kontak masuk	85
GAMBAR 3.24. <i>Activity Diagram</i> Mengelola kontak keluar	86
GAMBAR 3.25. <i>Activity Diagram</i> hapus kontak keluar	87
GAMBAR 3.26. <i>Activity Diagram</i> Mengelola kontak terkirim	88
GAMBAR 3.27. <i>Activity Diagram</i> hapus kontak terkirim	89
GAMBAR 3.28. <i>Sequence Diagram</i> login.....	90
GAMBAR 3.29. <i>Sequence Diagram</i> pengiriman pesan.....	91
GAMBAR 3.30. <i>Sequence Diagram</i> pengiriman pesan tersiar.....	92
GAMBAR 3.31. <i>Sequence Diagram</i> mengelola pesan terjadwal	93
GAMBAR 3.32. <i>Sequence Diagram</i> mengelola kontak	94
GAMBAR 3.33. <i>Sequence Diagram</i> mengelola kontak grup	95
GAMBAR 3.34. <i>Sequence Diagram</i> mengelola kontak masuk	96
GAMBAR 3.35. <i>Sequence Diagram</i> mengelola kontak keluar	97
GAMBAR 3.36. <i>Sequence Diagram</i> mengelola kontak terkirim	98
GAMBAR 3.37. <i>Class diagram</i>	99
GAMBAR 3.38. <i>Entity Relationship Diagram</i>	100
GAMBAR 3.39. Relasi antar tabel.....	101
GAMBAR 3.40. <i>Design interface</i> Form Login.....	105
GAMBAR 3.41. <i>Design interface</i> Dashboard.....	106
GAMBAR 3.42. <i>Design interface</i> Tulis pesan.....	107
GAMBAR 3.43. <i>Design interface</i> Kontak masuk.....	107
GAMBAR 3.44. <i>Design interface</i> Kontak keluar	108
GAMBAR 3.45. <i>Design interface</i> Kontak Terkirim.....	108
GAMBAR 3.46. <i>Design interface</i> Pesan Tersiar	109

GAMBAR 3.47. <i>Design interface</i> daftar kontak.....	109
GAMBAR 3.48. <i>Design interface</i> Daftar Grup	110
GAMBAR 3.49. <i>Design interface</i> Pesan Terjadwal	111
GAMBAR 3.50. Tampilan implementasi <i>form login</i>	112
GAMBAR 3.51. Tampilan implementasi <i>Dashboard</i>	113
GAMBAR 3.52. Tampilan implementasi tulis pesan.....	113
GAMBAR 3.53. Tampilan implementasi Kontak Masuk.....	114
GAMBAR 3.54. Tampilan implementasi Kontak Keluar	115
GAMBAR 3.55. Tampilan implementasi kontak Terkirim	115
GAMBAR 3.56. Tampilan implementasi Pesan Tersiar	116
GAMBAR 3.57. Tampilan implementasi daftar kontak	117
GAMBAR 3.58. Tampilan implementasi daftar grup.....	117
GAMBAR 3.59. Tampilan implementasi Pesan Terjadwal	118
GAMBAR 3.60. Tampilan implementasi mengirim pesan ke <i>customer</i>	119

DAFTAR TABEL

TABEL: 2.1. Bagan Alir Dokumen (<i>FlowMap</i>).....	27
TABEL: 2.2. Simbol pada <i>Use Case Diagram</i>	29
TABEL: 2.3. Simbol <i>Activity Diagram</i>	30
TABEL: 2.4. Simbol <i>Class Diagram</i>	32
TABEL: 2.5. Simbol <i>Sequence Diagram</i>	33
TABEL: 2.6. Simbol ERD notasi Chen	36
TABEL: 3.1. Kebutuhan Minimum Perangkat Keras	47
TABEL: 3.2. Kebutuhan Minimum Perangkat Lunak	48
TABEL: 3.3. Deskripsi perancangan <i>Use Case Admin</i>	49
TABEL: 3.4. Skenario <i>Use Case login</i>	53
TABEL: 3.5. Skenario <i>Use Case</i> pengiriman pesan	54
TABEL: 3.6. Skenario <i>Use Case</i> pengiriman pesan tersiar	54
TABEL: 3.7. Skenario <i>Use Case</i> mengelola pesan terjadwal.....	55
TABEL: 3.8. Skenario <i>Use Case</i> tambah pesan terjadwal.....	55
TABEL: 3.9. Skenario <i>Use Case</i> edit pesan terjadwal	56
TABEL: 3.10. Skenario <i>Use Case</i> hapus pesan terjadwal	56
TABEL: 3.11. Skenario <i>Use Case</i> mengelola kontak.....	57
TABEL: 3.12. Skenario <i>Use Case</i> tambah kontak.....	57
TABEL: 3.13. Skenario <i>Use Case</i> cari kontak	58
TABEL: 3.14. Skenario <i>Use Case</i> edit kontak	58
TABEL: 3.15. Skenario <i>Use Case</i> hapus kontak	59
TABEL: 3.16. Skenario <i>Use Case</i> mengelola kontak grup	59
TABEL: 3.17. Skenario <i>Use Case</i> tambah grup	60
TABEL: 3.18. Skenario <i>Use Case</i> cari grup.....	60

TABEL: 3.19. Skenario <i>Use Case</i> edit grup	61
TABEL: 3.20. Skenario <i>Use Case</i> hapus grup.....	61
TABEL: 3.21. Skenario <i>Use Case</i> mengelola kontak masuk	62
TABEL: 3.22. Skenario <i>Use Case</i> balas kontak masuk.....	62
TABEL: 3.23. Skenario <i>Use Case</i> hapus kontak masuk.....	63
TABEL: 3.24. Skenario <i>Use Case</i> mengelola kontak keluar	63
TABEL: 3.25. Skenario <i>Use Case</i> hapus kontak keluar	64
TABEL: 3.26. Skenario <i>Use Case</i> mengelola kontak terkirim.....	64
TABEL: 3.27. Skenario <i>Use Case</i> hapus kontak terkirim	65
TABEL: 3.28. Tabel <i>customer service</i>	102
TABEL: 3.29. Tabel kontak.....	102
TABEL: 3.30. Tabel kontak grup	102
TABEL: 3.31. Tabel pesan terjadwal.....	102
TABEL: 3.32. Tabel kontak masuk	103
TABEL: 3.33. Tabel kontak keluar.....	103
TABEL: 3.34. Tabel kontak terkirim.....	104
TABEL: 3.35. Rencana Pengujian aplikasi.....	120
TABEL: 3.36. Pengujian perancangan aplikasi	121

BAB I

PENDAHULUAN

1.1. Latar Belakang

Era teknologi informasi ditandai dengan kemudahan dan kecepatan aliran informasi dari satu komunitas ke yang lainnya. Salah satu produk teknologi informasi adalah telekomunikasi menggunakan perangkat *Handphone* atau telepon *seluler*. Di dalamnya terdapat sebuah metoda komunikasi yang dinamakan SMS (*Short Message Service*) yaitu sebuah konsep pengiriman informasi berbasis text.

Seiring dengan perkembangannya, SMS tidak hanya dipergunakan untuk mengirimkan atau bertukar informasi antara dua orang yang saling mengenal atau membutuhkan, SMS juga sudah mulai dipergunakan untuk berhubungan antara seseorang dengan sebuah *system* sesuai dengan kebutuhan (Dewanto R A. 2007: 35).

PT. Jaya Angkasa Globalindo merupakan perusahaan yang sangat profesional dan berpengalaman serta konsisten memposisikan diri sebagai spesialis kontraktor penangkal petir. Kami fokus menangani berbagai macam permasalahan yang disebabkan oleh ancaman bahaya sambaran petir di Indonesia. Wilayah tropis sebagaimana Indonesia sangatlah

dikenal sebagai wilayah yang memiliki curah petir harian atau intensitas terbesar didunia.

Permasalahan umum yang di alami PT. Jaya Angkasa Globalindo saat ini, dalam mengirim dan menerima SMS informasi dan keluhan kepada perusahaan dan terkait pelayanan yang dilakukan masih kurang cepat, karena masih manual dalam segi informasi, contohnya pada saat *customer service* mengirim sebuah SMS kepada *customer* yang jumlahnya puluhan masih secara manual dan memakan waktu yang lama, maka oleh karena itu perlu diterapkan strategi baru dengan menggunakan SMS *Gateway* untuk meningkatkan layanan yang maksimal kepada *customer*.

Berdasarkan permasalahan diatas, maka dibangunlah sebuah aplikasi untuk *customer service* berbasis *web* dengan PHP dan MySQL menggunakan SMS *Gateway*. Maka dari itu penulis mengambil judul **“PERANCANGAN APLIKASI UNTUK *CUSTOMER SERVICE* BERBASIS *WEB* MENGGUNAKAN SMS *GATEWAY* (Studi Kasus PT. JAYA ANGKASA GLOBALINDO)”**.

1.2. Identifikasi Masalah

Setelah meninjau meneliti dan mengikuti kegiatan-kegiatan yang ada di PT. Jaya Angkasa Globalindo, Akhirnya penulis menemukan beberapa kendala yang timbul pada PT. Jaya Angkasa Globalindo, kendala tersebut yaitu :

1. Bagaimana membangun aplikasi untuk memberikan informasi kepada para *customer* ?
2. Bagaimana mengatasi pelayanan terhadap pada *customer* yang dirasa masih kurang maksimal ?

1.3. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

1. Memangfaatkan aplikasi SMS *Gateway* untuk memberkan informasi kepada *customer* secara cepat dan efisien.
2. Dengan memanfaatkan aplikasi SMS *gateway* ini di harapkan bisa memaksimalkan pelayanan informasi kepada *customer*.

1.4. Batasan Masalah

Agar masalah yang di teliti tidak keluar dan menyimpang, maka diperlukan adanya suatu batasan masalah. Adapun batasan masalah tersebut adalah :

1. Obyek penelitian ini di PT. Jaya Angkasa Globalindo.
2. Dalam pembuatan SMS *Gateway* ini menggunakan *platrom web* dengan bahasa pemograman PHP dan MySQL.
3. Aplikasi ini dapat di jalankan di windows 7 32bit.
4. Didalam aplikasi ini terdapat fitur : pengiriman pesan, pengiriman pesan tersiar dan pesan terjadwal.
5. Aplikasi ini menggunakan service gammu.

6. Aplikasi ini digunakan oleh *customer service* untuk memberikan sebuah informasi kepada *customer* dan juga keluhan dari *customer* ke pada perusahaan.
7. Penelitian ini tidak sampai tahap *maintenance*.

1.5. Metode Penelitian

1.5.1. Teknik Pengumpulan Data

Adapun teknik pengumpulan data yang dilakukan penulis adalah :

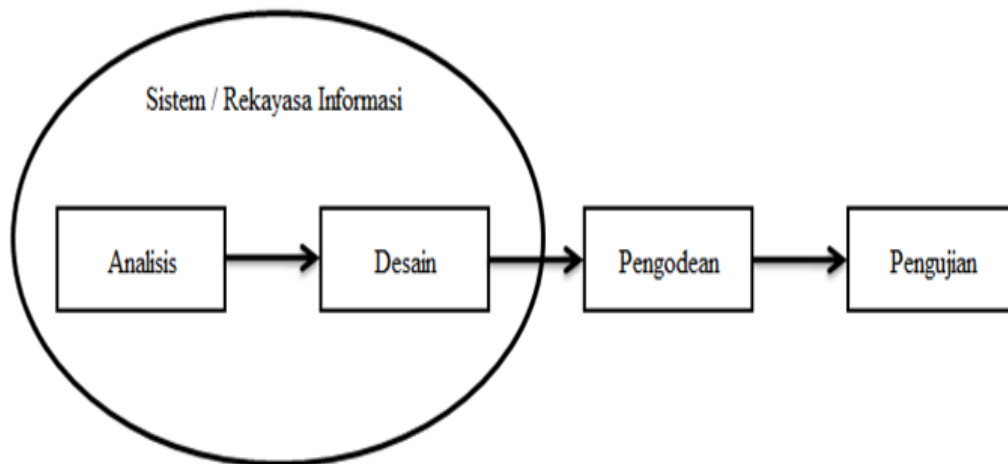
1. Studi Lapangan meliputi:
 - a. Observasi langsung yaitu mengadakan pengamatan secara langsung bagaimana kondisi dan sistem yang ada di PT. Jaya Angkasa Globalindo.
 - b. Wawancara yaitu teknik pengumpulan data dengan cara bertanya langsung kepada pihak-pihak yang berkaitan.

2. Studi Literatur:

Pada tahap ini dilakukan studi literatur tentang informasi terkait, yakni melakukan studi keperpustakaan terhadap referensi yang berkaitan dengan penelitian.

1.5.2. Metode Perancangan

Model SDLC (*System Development Lyfe Cycle*) air terjun (*waterfall*) bisa disebut juga model sekuensial linear (*Sequential linear*) atau alur hidup klasik (*classic life cycle*). Model *waterfall* adalah model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut yang dimulai dari analisis, desain, pengodean dan pengujian (Rosa A.S dan M.Shalahuddin, 2018). Berikut adalah gambar 1.1 model *waterfall*, yang mempunyai tahapan-tahapan sebagai berikut :



GAMBAR: 1.1. Ilustrasi Model *Waterfall* (Rosa A.S dan M.Shalahuddin, 2018)

Keterangan pada Gambar 1.1 diatas sebagai berikut:

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan

mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

1.6. Sistematika Penulisan

Untuk memberikan gambaran yang jelas mengenai penyusunan Tugas Akhir ini, maka ditetapkan sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang, identifikasi masalah, tujuan penelitian, batasan masalah, metode penelitian serta sistematika penulisan.

BAB II LANDASAN TEORI

Pada bab ini akan dijelaskan beberapa teori yang mendasari pemecahan masalah dan teori-teori yang berkaitan langsung dengan pemecahan masalah serta pembahasannya yang diperoleh dari buku-buku referensi dan catatan-catatan lainnya yang dikembangkan.

BAB III PEMBAHASAN

Bab ini berisi tentang analisis sistem yang meliputi analisis perusahaan, analisis prosedur yang sedang berjalan, analisis permasalahan, analisis sistem yang diusulkan, dan analisis perangkat yang akan digunakan, desain aplikasi yang meliputi : *use case diagram, activity diagram, sequence diagram, class diagram,*

Entity Relationship Diagram (ERD), pengkodean aplikasi dan sampai dengan tahap pengujian.

BAB IV KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan dan saran-saran dari hasil analisis dan pembahasan yang dapat diajukan sehubungan dengan permasalahan tugas akhir.

BAB II

LANDASAN TEORI

2.1. Pengertian Aplikasi

Menurut Nazrudin Safaat H (2012:9) perangkat lunak aplikasi adalah suatu *sub* kelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya perangkat lunak aplikasi dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna.

Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media. Beberapa aplikasi yang digabung bersama menjadi suatu paket kadang disebut sebagai suatu paket atau paket aplikasi (*application suite*). Contohnya adalah *Microsoft Office* dan *Open Office.org*, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan setiap aplikasi. Seringkali, aplikasi ini memiliki

kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah.

2.2. Definisi Sistem

Banyak orang yang tidak asing mendengar istilah sistem, seringkali mereka berpendapat bahwa sistem mengacu pada komputer. Namun pada kenyataannya sistem memiliki pengertian yang lebih luas lagi. Berikut ini beberapa pengertian sistem menurut para Ahli :

“Sistem adalah suatu rangkaian yang terdiri dari dua atau lebih komponen yang saling berhubungan dan saling berinteraksi satu sama lain untuk mencapai tujuan dimana sistem biasanya terbagi dalam sub sistem yang lebih kecil yang mendukung sistem yang lebih besar” (Romney dan Steinbart, 2015:3).

“Sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Sebagai gambaran jika dalam sebuah sistem terdapat sebuah elemen yang tidak memberikan manfaat dalam mencapai tujuan yang sama maka elemen tersebut dapat dipastikan bukanlah bagian dari sistem” (Abdul Kadir, 2014:61).

“Sistem adalah kumpulan/grup dari sub sistem/bagian/komponen apapun baik fisik ataupun non fisik yang saling berhubungan satu sama

lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu” (Azhar Susanto, 2013:22).

Dari pengertian diatas dapat disimpulkan bahwa sistem merupakan kumpulan dari beberapa komponen yang saling berhubungan dan bekerja dengan teratur dan harmonis untuk mencapai tujuan tertentu. Terdapat dua kelompok pendekatan dalam mendefinisikan sistem. Tata Sutabri (2012) yaitu :

1. Pendekatan sistem yang lebih menekankan pada prosedur, mendefinisikan sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan. Berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.
2. Pendekatan yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai suatu kumpulan dari elemen-elemen yang saling berinteraksi untuk mencapai suatu tujuan tertentu.

Berdasarkan beberapa pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa sistem adalah suatu kumpulan bagian-bagian baik manusia atau pun bukan manusia yang saling berinteraksi untuk mencapai suatu tujuan.

2.2.1. Karakteristik Sistem

Model umum sebuah sistem adalah *input*, proses, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu,

sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud oleh Tata Sutabri (2012) adalah sebagai berikut:

1. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi yang berarti saling bekerjasama membentuk satu kesatuan. Komponen dari sistem dapat disebut subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi suatu proses sistem dalam keseluruhan.

2. Batasan Sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya atau lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*Scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar (*Environment System*) dari suatu sistem adalah apapun yang ada diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan ataupun sebaliknya merugikan. Lingkungan luar sistem yang menguntungkan merupakan energi dari sistem maka harus tetap dijaga dan dipelihara,

sedangkan lingkungan luar sistem yang merugikan harus ditahan atau dikendalikan agar tidak mengganggu kelangsungan hidup sistem.

4. Penghubung sistem (*Interface*)

Penghubung sistem merupakan media yang menghubungkan antara subsistem satu dengan subsistem lainnya. Keluaran (*Output*) dari suatu subsistem akan menjadi masukan (*Input*) untuk subsistem lainnya dengan melalui penghubung. Maka dengan penghubung satu sistem dapat berinteraksi dengan subsistem lainnya untuk membentuk satu kesatuan.

5. Masukan Sistem (*Input*)

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa perawatan (*Maintenance Input*) masukan sinyal (*Signal Input*). *Maintenance input* adalah energi yang dimasukkan agar sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses agar dihasilkan sebuah keluaran (*Output*). Sebagai contoh di dalam sistem komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan data adalah *signal input* yang diolah menjadi sistem informasi.

6. Keluaran Sistem (*Output*)

Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan berupa sisa pembuangan. Keluaran dapat menjadi masukan untuk subsistem yang lainnya atau kepada sistem. Misalnya pada sistem komputer, sebagai keluaran yang

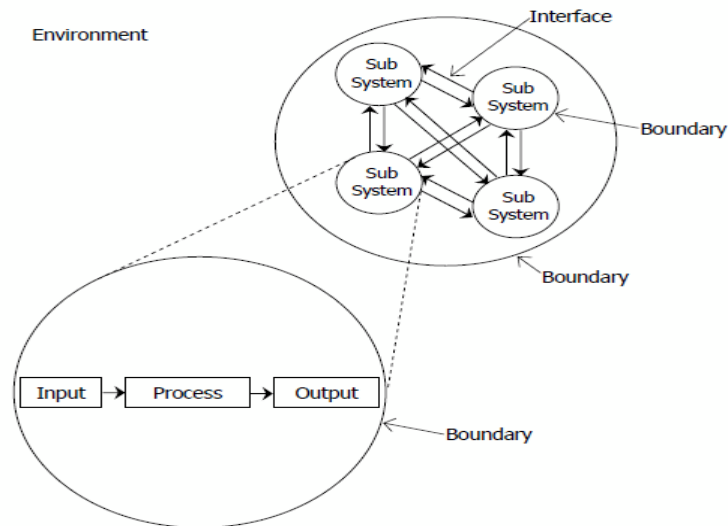
berguna adalah informasi yang dibutuhkan, kemudian sebagai keluaran yang tidak dibutuhkan adalah panas yang dihasilkan.

7. Pengolahan Sistem (*Process*)

Suatu sistem dapat memiliki suatu bagian pengolahan. Bagian pengolahan ini yang akan mengubah masukan mejadi keluaran. Misalnya pada sistem produksi akan mengolah masukan berupa bahan baku dan barang-barang lainnya menjadi barang jadi.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan saran yang pasti dan bersifat *deterministic*. Jika suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenal sasaran atau tujuan yang telah direncanakan.



GAMBAR: 2.1. Karakteristik sistem (Tata Sutabri, 2012)

2.3. *Customer Service*

Menurut Heru Susanto (2013:116-117) *Customer service* adalah suatu tugas lain dari kegiatan penjualan secara aktif, tugasnya termasuk dalam hubungan langsung dengan konsumen itu sendiri, atau melalui alat komunikasi, surat atau proses otomatis. *Customer service* dirancang, dilaksanakan dan dikomunikasikan dengan dua tujuan utama, yaitu produktivitas operasional dan kepuasan konsumen.

Pengertian *customer service* secara umum adalah “setiap kegiatan yang diperuntukkan atau ditujukan untuk memberi kepuasan kepada nasabah, melalui pelayanan yang dapat memenuhi keinginan dan kebutuhan nasabah”.

Menurut Philip Kotlet “pelayanan atau *service* adalah setiap kegiatan atau manfaat yang dapat diberikan suatu pihak kepada pihak lain yang pada dasarnya tidak berwujud dan tidak pula berakibat pemilikan sesuatu dan produksinya dapat atau tidak dapat dikaitkan dengan suatu produksi fiksi”.

Dari uraian diatas dapat ditarik kesimpulan bahwa *customer service* adalah orang yang bertugas untuk memberikan pelayanan informasi dan merupakan perantara antara bank dengan nasabah yang akan menggunakan produksi bank tersebut, serta memberikan keuntungan bagi kedua belah pihak. Seorang *customer service* diberikan beban dalam bertanggung jawab dari awal sampai selesai dalam melayani nasabah secara prima.

2.4. *Short Massaging Services* (SMS)

SMS adalah kepanjangan dari *Short Massaging Services* (Layanan Pesan Pendek). Di zaman yang serba modern ini, hampir semua orang pernah menggunakannya. SMS adalah sebuah bentuk layanan dari penyedia jasa layanan telekomunikasi atau *provider* telekomunikasi. Sebuah pesan elektronik yang dikirimkan melalui media telepon seluler atau telepon genggam (*handphone*) yang kemudian diterima oleh perangkat yang sama (*receiver*) berupa telepon juga (Sofyan Maulana, 2015:1).

2.4.1. Karakteristik SMS

Menurut Wicksono (2007:3) ada beberapa karakteristik SMS yaitu:

1. sebuah pesan singkat terdiri atas 160 karakter yang mencakup huruf dan angka. Juga dapat mendukung pesan *non-teks*, seperti format *binary*.
2. Prinsip kerjanya adalah “menyimpan” dan “menyampaikan” pesan (*store and forward message*). Dengan kata lain, pesan tidak langsung dikirim ke penerima melainkan disimpan terlebih dahulu di *SMS-Center*.
3. Memiliki ciri-ciri dalam konfirmasi pengiriman pesan, yaitu pesan yang dikirimkan tidak secara sederhana dikirimkan dan dipercayai akan disampaikan dengan selamat. Namun, pengiriman pesan tidak pula menerima pesan balik yang memberitahu apakah pesan telah terkirim atau gagal.

2.4.2. SMS Center (SMSC)

Sebuah SMSC bertanggung jawab untuk menangani sebuah pesan SMS pada jaringan nirkabel (*wireless*). Ketika sebuah SMS dikirimkan melalui telepon selular, SMS tersebut pertama kali akan di tampung oleh SMSC, kemudian akan diteruskan ke nomor tujuan. Sistem ini lebih dikenal dengan sebutan *store and forward*. Hal ini berarti pesan akan di tampung dalam SMSC terlebih dahulu sebelum diteruskan ke nomor tujuan. Apabila nomor tujuan sedang tidak aktif, maka pesan akan tetap disimpan dalam SMSC sampai masa aktif pesan tersebut habis. Terkadang sebuah pesan SMS harus melewati beberapa jaringan yang berbeda (SMSC maupun SMS Gateway) sebelum sampai ke tujuan (Wicksono, 2007:4).

2.4.3. Manfaat Pengguna SMS

Menurut Wicksono (2007:5) Penggunaan SMS secara tidak langsung telah mendominasi sebagian besar pengguna telepon seluler di dunia. Tentunya hal ini diiringi dengan adanya manfaat maupun keunggulan dari SMS itu sendiri.

Pada dasarnya SMS mempunyai manfaat sebagai berikut :

1. menerima dan mengirim pesan, baik itu pesan *standart*, notifikasi dan lain-lain.
2. Maupun mengirimkan pesan ke banyak nomor tujuan pada waktu yang sama.

3. Memiliki tingkat kegagalan kirim yang sangat kecil sehingga pesan kemungkinan besar akan sampai pada tujuan.
4. Merupakan sebuah mekanisme komunikasi yang tergolong membutuhkan biaya yang murah.

2.5. Pengertian SMS Gateway

SMS Gateway adalah suatu *platform* yang menyediakan mekanisme untuk mengirim dan menerima SMS. SMS Gateway dapat berkomunikasi dengan perangkat lain yang memiliki SMS *platform* untuk menghantarkan dan menerima pesan SMS dengan sangat mudah. Hal ini memungkinkan karena SMS Gateway juga dibekali tampilan antarmuka yang mudah dan *standart* (Wahana Komputer, 2014:1-2).

Pengguna SMS Gateway ini memungkinkan sudah sering anda jumpai namun tidak anda sadari. Misalnya pada aplikasi *web* yang telah menggunakan SMS sebagai media konfirmasi keanggotaan, konter pulsa, acara pengundian di televisi, layanan SMS premium, pemberitahuan dari *provider* layanan komunikasi.

Fitur yang bisa dijumpai pada SMS Gateway, antara lain:

1. Auto Reply

Fitur yang biasa dijumpai pada program “ketik REG”. pada program tersebut anda wajibkan mendaftar terlebih dahulu sebelum dapat menikmati layanan yang ditawarkan format SMS pada saat mendaftar

juga telah di tetapkan. Jika format SMS yang dikirim salah, anda akan mendapat balasan bahwa format yang anda kirimkan salah.

2. Pengiriman masal/*broadcast message*

Fitur pengiriman masal ini sudah banyak disediakan pada model *handphone* sekarang. Namun pada fitur tersebut masih terdapat kekurangan, yaitu SMS yang dikirim haruslah sama. Jika anda ingin mengirim SMS ke banyak orang sekaligus menyebutkan nama merk di dalam SMS anda, hal tersebut tidak dapat dilakukan. Dengan aplikasi *SMS Gateway* hal tersebut dapat dilakukan.

3. Pengiriman terjadwal

Fitur ini memungkinkan anda mengirim SMS pada waktu yang telah di tetapkan sebelumnya. Fitur ini biasa digunakan pada upacara selamat ulang tahun maupun sebagai pesan singkat.

Menurut Saptaji (2012:11) menjelaskan bahwa “*Short Message Service* (SMS) atau layanan pesan singkat merupakan suatu teknologi yang memungkinkan perangkat seluler untuk mengirim dan menerima pesan dalam bentuk teks”.

Awal mulanya teknologi ini digunakan dalam teknologi GSM (*Global System for Mobile Communication*), namun perkembangan selanjutnya juga diadopsi teknologi CDMA (*Code Division Multiple Access*) dan UMTS/3G.

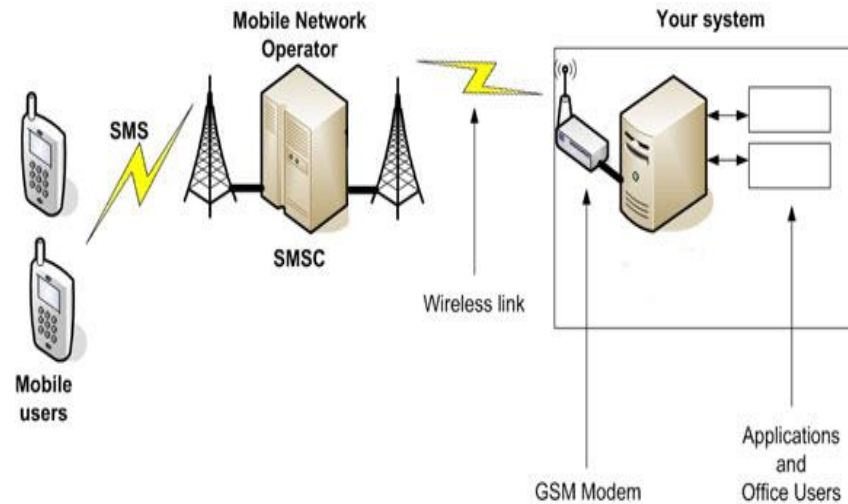
Dalam jaringan GSM, SMS ditangani oleh sebuah perangkat yang bernama SMSC (*SMS Center*) yang berinteraksi dengan HLR (*Home*

Location Register) untuk mencari database dan profil pelanggan, dan MSC (*Mobile Switching Center*) untuk melakukan proses *switching* dan *routing* ke pelanggan.

Dalam terminologi teknologi SMS, ada dua proses penanganan SMS ketika SMS itu dimulai dikirim sampai diterima oleh nomor tujuan. Saat SMS dikirim terjadi proses SMS-MO (*SMS Mobile Originating*) yakni proses dari MS (*Mobile Station*) sampai dengan SMSC, dan proses penyampaian SMS dari SMSC ke MS tujuan yang disebut SMS-MT (*SMS Mobile Terminating*).

Saptaji (2012:13) menjelaskan bahwa “*SMS Gateway* adalah sebuah perangkat atau sistem yang mampu menangani operasi dasar SMS (mengirim, menerima, membaca dan menghapus SMS) yang berada diluar jangkauan GSM standar”.

Dalam SMS Gateway juga dapat ditambahkan fitur pelengkap seperti *Autoreply*, *SMS Broadcast*, *SMS Scheduler*, *SMS Polling*, dll. *SMS Gateway* berwujud sebuah perangkat *kontroller* (misalnya komputer) dan perangkat seluler (misal *handphone* atau modem) sebagai penghubung ke jaringan SMS.



GAMBAR: 2.2. Alur kerja SMS Gateway (Ardana, 2004:35)

2.6. Gammu

Gammu adalah semacam *service* yang disediakan untuk membangun aplikasi SMS Gateway. *Gammu* merupakan media penghubung antara komputer dengan perangkat telepon untuk dapat digunakan mengirim/menerima SMS. Selain SMS, *Gammu* juga dapat digunakan untuk mengirim/menerima MMS dan *backup/restore phonebook* (Wahana komputer, 2014:3).

Menurut sofyana maulana (2015:11-12) Kelebihan *Gammu* dibandingkan *tools SMS gateway* lainnya adalah :

1. *Gammu* bias di jalankan di *Windows* maupun *linux*.
2. Banyak *device* yang kompatibel dengan *gammu*.
3. *Gammu* menggunakan *database MySQL*.
4. Banyak kabel data USB maupun serial semuanya kompatibel dengan *Gammu*.

5. *Gammu* adalah aplikasi *Open Source* yang dapat dipakai secara gratis.
6. *Gammu* tidak memerlukan banyak *hardware* (hanya perlu PC dan modem/*handphone*) sehingga memudahkan pengembangan aplikasi dengan modal sedikit.

2.7. Unsur-Unsur Perancangan Website

Ada beberapa hal unsur-unsur dalam sebuah perancangan *website* diantaranya terdiri dari HTML, *Web Browser*, *Web Server*, PHP, SQL, CSS dan *CodeIgniter* :

2.7.1. HTML

HTML adalah *Hypertext Markup Language* yang artinya adalah sebuah teks berbentuk *link* dan mungkin juga foto atau gambar yang saat di klik, akan membawa si pengakses internet dari satu dokumen ke dokumen lainnya. Dalam praktiknya, *Hypertext* berwujud sebuah *link* yang bias mengantar anda ke dunia internet yang sangat luas. Untuk membantu si pengakses berpindah dari satu tempat ke tempat lainnya, dibuatlah semacam dokumen yang nanti akan disebut dengan istilah *website*. Untuk membuat *website*, kita membuat *Markup*, yaitu *Tag* semacam kode yang mengatur bagaimana *website* tersebut akan ditampilkan di jendela *browser*, seperti *layout* dan tampilan-tampilan visual yang biasa kita lihat di dalam sebuah *website* (Jubilee, 2016:16).

2.7.2. Web Browser

Web adalah suatu metode untuk menampilkan informasi di internet, baik berupa teks, gambar, suara maupun video yang interaktif dan mempunyai kelebihan untuk menghubungkan (*link*) satu dokumen dengan dokumen lainnya (*hypertext*) yang dapat diakses melalui sebuah *browser* (Yuhefizar 2009:2).

2.7.3. Web Server

Web server adalah sebuah perangkat lunak *server* yang berfungsi menerima permintaan HTTP dan HTTPS dari klien yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya dalam halaman-halaman yang umumnya berbentuk dokumen HTML (Kurniawan 2008:2).

2.7.4. PHP

PHP merupakan secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen HTML secara *On the fly* yang dieksekusi di *server web*, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor *teks* atau editor HTML. Dikenal juga sebagai bahasa pemrograman *server side*. 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 (Betha Sidik, 2017:4).

2.7.5. SQL

Menurut Eri Mardiani (2016:13) SQL merupakan kependekan dari kata “*Structured Query Language*”. SQL merupakan suatu bahasa permintaan yang terstruktur. Dikatakan terstruktur karena pada penggunaannya, SQL memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama ANSI.

SQL (*Structured Query Language*) adalah sebuah bahasa yang dipergunakan untuk mengakses data dalam basis data relasional. Bahasa ini secara *de facto* merupakan Bahasa *standart* yang digunakan dalam manajemen basis data relasional. saat ini, hampir semua *server* basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya.

2.7.6. *Cascade Sheet Style (CSS)*

Menurut Rintho (2018:133) CSS adalah singkatan dari *Cascading Style Sheets*. CSS Biasanya selalu dikaitkan dengan HTML, karena keduanya memang saling melengkapi dimana HTML ditujukan untuk membuat struktur, atau konten dari halaman *web*. Sedangkan CSS digunakan untuk tampilan dari halaman *web* tersebut. Istilahnya “*HTML for content, CSS for Presentation*”.

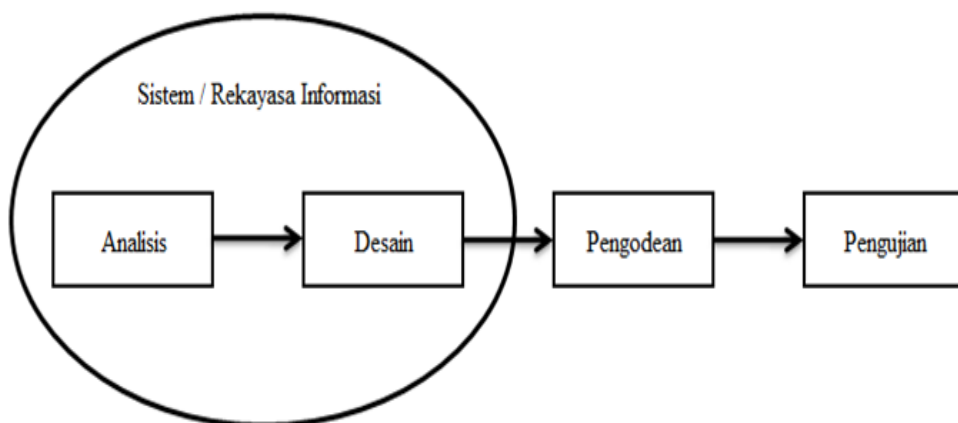
2.7.7. *CodeIgniter*

CodeIgniter adalah sebuah *framework* bahasa pemrograman PHP. *CodeIgniter* bisa dibilang *framework* PHP paling populer di Indonesia berkat kemudahan yang ditawarkan dalam penggunaannya (Wahana Komputer, 2014:4).

CodeIgniter menawarkan kemudahan serta standarisasi dalam proses pengembangan *website* dan aplikasi berbasis *web*. Dengan *CodeIgniter* proses pengembangan *website* menjadi lebih cepat dan terstandar. Standar ini sangat berguna dalam pengembangan *website* yang dikerjakan oleh tim. *CodeIgniter* juga telah menyediakan *library* dan *helper* yang digunakan dan mempermudah proses *development*.

2.8. Metode *Waterfall*

Model SDLC (*System Development Lyfe Cycle*) air terjun (*waterfall*) bisa disebut juga model sekuensial linear (*Sequential linear*) atau alur hidup klasik (*classic life cycle*). Model *waterfall* adalah model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut yang dimulai dari analisis, desain, pengodean dan pengujian (Rosa A.S dan M.Shalahuddin, 2018).



GAMBAR: 2.3. Ilustrasi Model *Waterfall* (Rosa A.S dan M.Shalahuddin, 2018)

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.


2.9. Diagram Alir Dokumen (*FlowMap*)



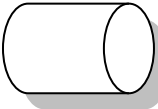
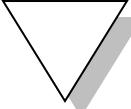
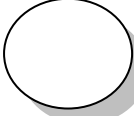
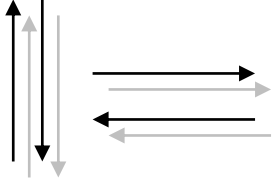
Definisi *flowmap* menurut Ladjamudin bin Al-Bahra adalah sebagai berikut :

“Flowmap adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. Flowmap merupakan cara penyajian dari suatu algoritma” (Al-Bahra Bin Ladjamudin, 2005:72-73).

Bagan alir dokumen atau disebut juga bagan alir formulir atau *paperwork flowmap* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir dan termasuk tembusan-tembusannya.

TABEL: 2.1 Bagan Alir Dokumen (*FlowMap*)

Simbol	Keterangan
	Dokumen, Menunjukkan dokumen berupa dokumen input dan output baik untuk proses manual maupun proses berbasis mekanik atau komputer.

	Proses manual, Menunjukkan proses yang dilakukan secara manual.
	Simbol Proses Terkomputerisasi, Menunjukkan kegiatan proses yang dilakukan secara komputerisasi.
	File Hardisk, Menunjukkan media penyimpanan data/informasi file pada proses berbasis komputer. File dapat disimpan pada lemari arsip, map dll.
	Pengarsipan (<i>Offlene Storage</i>), Menunjukkan simpanan data non komputer/informasi file pada proses manual.
	Penghubung, Menunjukkan alir dokumen yang terputus atau terpisah pada halaman alir dokumen yang sama.
	Arah Alir Dokumen, Menunjukkan arah aliran dokumen antar bagian yang terkait pada suatu sistem. Dapat dari sistem keluar atau pun dari luar ke sistem dan antar bagian diluar sistem.

2.10. *Unified Modeling Language (UML)*

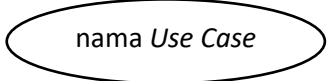



UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa A.S dan M. Salahuddin, 2018:133). Berikut adalah beberapa diagram yang dimasukkan dalam sistem sebagai berikut :

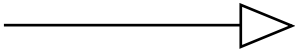
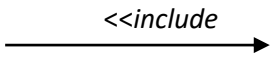
2.10.1. Use Case Diagram

Use Case Diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa aja yang berhak menggunakan fungsi-fungsi itu (Rosa A.S dan M. Salahuddin, 2018:155).

Tabel 2.2 menjelaskan tentang simbol *use case diagram*.

TABEL: 2.2. Simbol pada *Use Case Diagram*


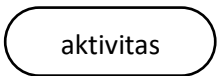
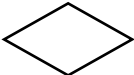
Simbol	Deskripsi
<p><i>Use Case</i></p> 	<p>Sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase</i> nama <i>use case</i>.</p>
<p>Aktor / <i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi. Aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama aktor.</p>
<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 interaksi dengan aktor.</p>
<p>Ekstensi / <i>extend</i></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; mirip prinsip <i>inheritance</i> pada pemograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>

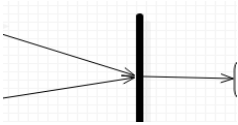

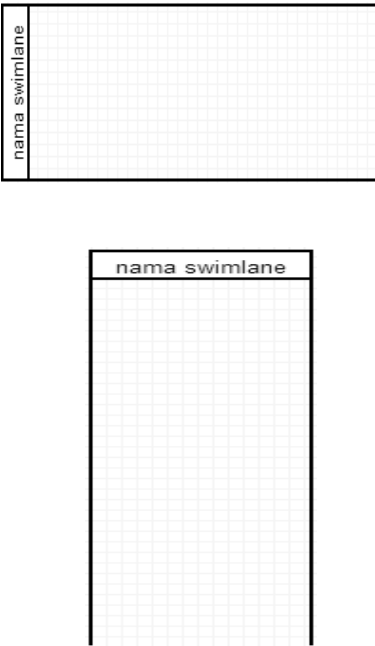
<p>Generalisasi / <i>generalisation</i></p> 	<p>Hubungan generalisasi dengan spesialisasi antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum.</p>
<p>Manggunakan <i>include</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i>.</p>

2.10.2. Activity Diagram

Diagram aktifitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa A.S dan M. Salahuddin, 2018:161). Tabel 2.3 menjelaskan tentang simbol *Activity diagram*.

TABEL: 2.3. **Simbol Activity Diagram**

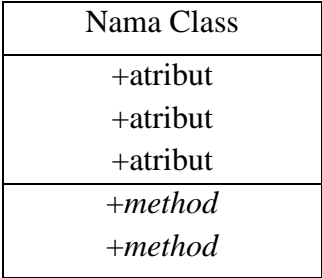

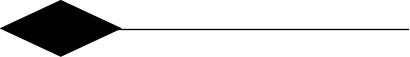
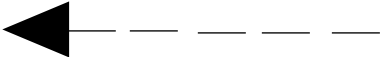

Simbol	Deskripsi
<p>Status awal</p> 	<p>Sebuah diagram aktivitas memiliki sebuah status awal.</p>
<p>Aktivitas</p> 	<p>Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.</p>
<p>Percabangan / <i>decision</i></p> 	<p>Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.</p>

<p>Penggabungan / <i>join</i></p> 	<p>Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.</p>
<p>Status akhir</p> 	<p>Sebuah diagram aktivitas memiliki sebuah status akhir.</p>
<p>Swimlane</p> 	<p>Untuk memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.</p>

2.10.3. Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Rosa A.S dan M. Salahuddin, 2018:141). Tabel 2.4 menjelaskan simbol pada *class diagram*.

TABEL: 2.4. **Simbol Class Diagram**


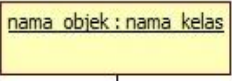

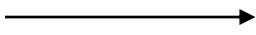


Simbol	Deskripsi
<p><i>class</i></p> 	<p><i>Class</i> adalah blok – blok pembangun pada pemrograman berorientasi obyek dan terdiri dari tiga bagian. Nama <i>class</i>, atribut dan <i>method</i>.</p>
<p>Asosiasi / <i>association</i></p> 	<p>Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>.</p>
<p><i>Composition</i></p> 	<p>Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>composition</i> terhadap <i>class</i> tempat dia bergantung tersebut.</p>
<p><i>Dependency</i></p> 	<p>Ketergantungan antar kelas.</p>
<p><i>aggregation</i></p> 	<p>Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>).</p>

2.10.4 *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek – objek yang terlibat dalam sebuah *use case* beserta metode – metode yang dimiliki kelas yang diinstiasiasi

menjadi objek itu (Rosa A.S dan M. Salahuddin, 2018:165). Tabel 2.5 menjelaskan simbol pada *sequence* diagram.

TABEL: 2.5. Simbol *Sequence Diagram*

Simbol	Deskripsi
<p>Aktor</p> 	Menggambarkan orang yang berinteraksi dengan sistem.
	Menggambarkan objek yang berinteraksi pesan.
<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya
<p>Pesan tipe send</p> <p>1: masukan</p> 	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah paada objek yang dikirim.
<p><i>Lifeline</i></p> 	Menyatakan kehidupan suatu objek.
<p>Pesan tipe <i>return</i></p> 	Menggambarkan pengiriman pesan.

2.11. Basis Data

Menurut Nur Elfi dan Yvonne Wangdra (2016:119) Basis data dan teknologinya telah memainkan peran penting seiring dengan pertumbuhan penggunaan komputer. Basis data telah digunakan hamper seluruh area dimana komputer digunakan termasuk bisnis, teknik, kesehatan, hukum, pendidikan, dan sebagainya. Kata basis dapat diartikan sebagai kumpulan data yang saling berhubungan. Sedangkan data dapat diartikan sebagai fakta yang direkam atau dicatat. Sehingga basis data sendiri dapat disimpulkan sebagai :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa tanpa pengulangan, untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file/table/arsip* yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

2.11.1. MySQL

MySQL merupakan *software database* yang termasuk paling populer di lingkungan *Linux*, kepopuleran ini karena si tunjang karena performasi *query* dari *database* yang saat itu bisa dikatakan paling cepat dan jarang bermasalah. MySQL telah tersedia di lingkungan *windows*.

Berangkat dari *software* yang *shareware* MySQL populer, kini mulai versi 3.23 MySQL menjadi *software open source* yang berarti *free*. MySQL dapat digunakan untuk kepentingan komersial atau pun personal (*non profit*) (Betha Sidik, 2017:301).

2.11.2. PHPMyAdmin

PHPMyAdmin adalah aplikasi PHP sebagai administrator MySQL. PHPMyAdmin mendukung berbagai aktivitas MySQL seperti pengelolaan data, *table*, relasi antar *table* dan lain sebagainya (Rahman Su, 2013:12).

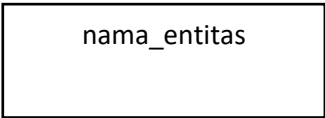
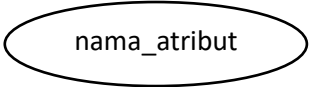
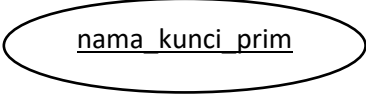
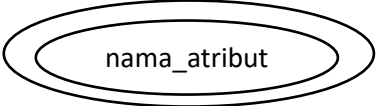
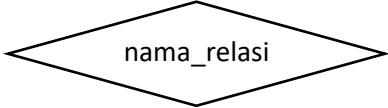
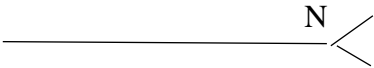
Tanpa program seperti PHPMyAdmin ini segala aktivitas *Query* pada *database* MySQL hanya bisa dilakukan oleh orang yang sudah mahir di bidangnya. Tapi kini dengan bantuan PHPMyAdmin mengelola sebuah *database* MySQL menjadi sangat mudah dan bisa dilakukan oleh pemula sekalipun.

2.12. Entity Relationship Diagram (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 relational .ERD memiliki beberapa aliran notasi seperti Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi *Crow's Foot*, dan beberapa notasi lain. Namun yang banyak digunakan

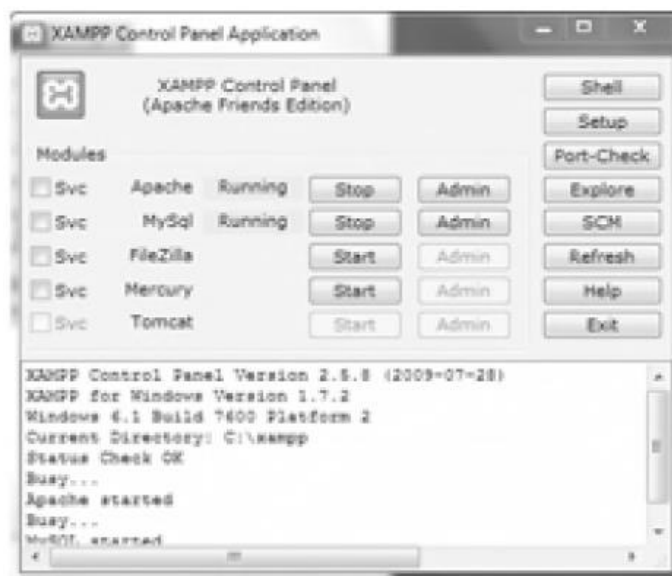
adalah notasi Chen (Rosa A.S dan M. Salahuddin, 2018:50). Berikut adalah simbol – simbol yang digunakan pada ERD dengan notasi Chen.

Tabel 2.6. Simbol ERD notasi Chen

Simbol	Deskripsi
Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan, bakal tabel pada basis data, benda yang memiliki data dan harus disimpandatanya agar dapat diakses oleh aplikasi komputer.
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam satu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan, biasanya berupa id.
Atribut multivalai/ <i>multivalue</i> 	<i>Field</i> ata kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi 	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja.
Asosiasi / <i>association</i> 	Penghubung antar relasi dan entitas dimana kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian.

2.13. XAMPP

Menurut Sidik (2014:72), bahwa “Xampp (X(Windows/Linux) Apache MySQL PHP dan Perl) merupakan paket *server web* PHP dan *database* MySQL yang paling populer dikalangan pengembang *web* dengan menggunakan PHP dan MySQL sebagai *databasenya*”. Dikatakan oleh Bekti (2015:30) Melalui program ini, *programmer web* dapat menguji aplikasi *web* yang dikembangkan dan mempresentasikannya ke pihak lain secara langsung dari komputer, tanpa perlu terkoneksi ke internet. Xampp juga dilengkapi fitur manajemen *database* phpMyAdmin seperti pada *server hosting* sungguhan, sehingga pengembang *web* dapat mengembangkan aplikasi *web* berbasis *database* secara mudah.



GAMBAR: 2.4. Control Panel Xampp

2.14. *Black Box Testing*

Pengujian menggunakan sekumpulan aktifitas validasi, dengan pendekatan *black box testing*. Menurut Rosa A.S dan M Salahuddin (2014), *black box testing* adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian *black box* dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian *black box testing* harus dibuat dengan kasus benar dan kasus salah.

BAB III

PEMBAHASAN

Dalam penelitian ini digunakan metode pengembangan SDLC (*System Development Life Cycle*) dengan Model *Waterfall* Rosa A.S dan M.Shalahuddin (2018:29). Ada lima tahap dalam pengembangannya namun dalam penelitian ini dibatasi sampai dengan tahap empat, keempat tahap tersebut yaitu analisis kebutuhan perangkat lunak, desain, pembuatan kode program, dan pengujian.

3.1. Analisis

Pada tahap ini dilakukan proses analisis perusahaan, analisis prosedur yang sedang berjalan, analisis permasalahan, analisis sistem yang diusulkan, dan analisis perangkat yang digunakan.

3.1.1. Analisis Perusahaan

JAG Group merupakan perusahaan yang sangat profesional dan berpengalaman serta konsisten memposisikan diri sebagai spesialis kontraktor penangkal petir. Perusahaan ini fokus menangani berbagai macam permasalahan yang di sebabkan oleh ancaman bahaya sambaran petir di Indonesia. Wilayah tropis sebagaimana Indonesia sangatlah dikenal

sebagai wilayah yang memiliki curah petir harian atau intensitas petir terbesar didunia.

Berawal dari CV. Jaya Abadi Globalindo yang di dirikan oleh Pirman Suharto pada tahun 2009, beliau merupakan pengusaha muda yang senang melakukan penelitian dalam industri penangkal petir, sehingga beliau selalu mengikuti perkembangan teknologi system proteksi petir terpadu yang berkembang di dunia. Kemudian pada tahun 2014 beliau mendirikan beberapa perusahaan yang bergerak di bidang industri penangkal petir, salah satunya PT. Jaya Angkasa Globalindo yang telah banyak memiliki rekanan dan telah menangani banyak project pemasangan instalasi penangkal petir di seluruh pelosok tanah air, mulai dari perkantoran, gedung apartemen, kantor pemerintahan, pasar swalayan, tempat ibadah, rumah sakit, showrooms, perhotelan, sekolah, pabrik dan pergudangan, padang golf, perkebunan, peternakan, menara pemancar, pertambangan, migas, bandara, taman rekreasi, dll.

Dengan sumber daya manusia yang berkompeten dalam bidangnya, beberapa perusahaan swasta sebagai instalatur penangkal petir di Indonesia banyak yang memposisikan JAG Group sebagai mitra kerja bahkan sebagai konsultan pemasangan instalasi penangkal petir. Sudah ribuan pemasangan instalasi penangkal petir serta hasil karya pengamanan yang merupakan solusi dari permasalahan sambaran petir perusahaan kami tersebar di seluruh bumi Indonesia.

Sebagai langkah awal, perusahaan ini telah memperoleh kepercayaan sebagai *Exclusive Dealer* untuk wilayah Indonesia atas produk penangkal petir *elektrostatik* terbaik dan berkualitas tinggi bagi daerah tropis dengan merk Penangkal Petir *Flash Vectron*. Pada tahap berikut, perusahaan ini menyuplai produk-produk terkait mengenai instalasi penangkal petir dengan berbagai macam jenis dan merk diantaranya : Penangkal Petir Konvensional (*Faraday Cage/Franklin Rod*), *Lightning Counter*, *Surge Arrester*, *Obstruction Lamp*, Kabel Penghantar, *Grounding System*, Tower Penyangga, dan lain-lain.

Dengan area yang luas di Indonesia serta semakin bertambah majunya teknologi dibidang elektronika, perusahaan ini sangat optimis memandang bahwa sistem perlindungan petir merupakan bidang yang sangat prospektif dan penuh tantangan di masa depan.

3.1.1.1. Visi PT. JAYA ANGKASA GLOBALINDO

Memberikan solusi petir yang terbaik dan berkualitas dalam memenuhi kebutuhan perlindungan aset, struktur bangunan dan jiwa manusia terhadap adanya bahaya sambaran petir yang mengancam, serta membantu pembangunan dengan merealisasikan karya kami di Indonesia.

3.1.1.2. Misi PT. JAYA ANGKASA GLOBALINDO

Membantu pembangunan dengan merealisasikan karya kami dalam bentuk perlindungan struktur bangunan yang terbaik di seluruh penjuru Indonesia, serta menjadikan perusahaan spesialis penangkal petir atau anti

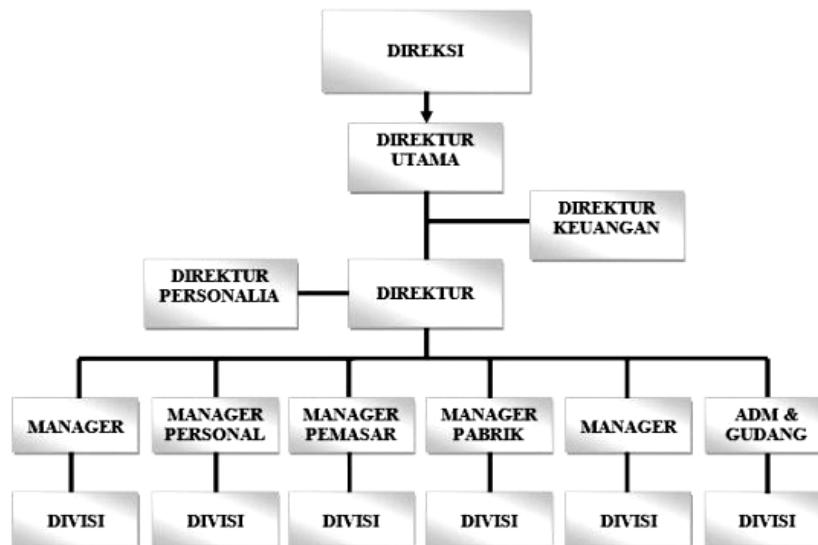
petir yang terdepan dan terbaik di Indonesia yang berteknologi modern sesuai perkembangan terkini.

3.1.1.3. Legalitas Perusahaan

PT. Jaya Angkasa Globalindo merupakan perusahaan yang bergerak dalam industri penangkal petir dan memiliki legalitas perusahaan sebagai berikut:

1. Akta Notaris, di terbitkan oleh Ibu Nenden Anne Handriyani, SH yang di register oleh Pengadilan Negeri Bandung.
2. SKT, NO. S-1261KT/WPJ.09/KP.0203/2015, dari Kementrian Keuangan Republik Indonesia Direktorat Jenderal Pajak
3. NPWP, NO. 72.261.624.0-423.000
4. TDP, NO. 101114621464 di terbitkan oleh Pemerintah Kota Bandung
5. IG/ITU, NO. 503/IG/DF27/BPPT
6. SIUP, NO. 510/1-CQ50/BPPT
7. Surat Keputusan MH & HAM RI, NO. AHU-0008461.AH.01.01.TAHUN 2015.
8. Rekening Perusahaan, atas nama PT. Jaya Angkasa Globalindo yang diterbitkan Bank Mandiri.
9. Dokumen lainnya sebagai pendukung.

3.1.1.4. Struktur Organisasi



GAMBAR: 3.1. Struktur Organisasi PT. Jaya Angka Globalindo

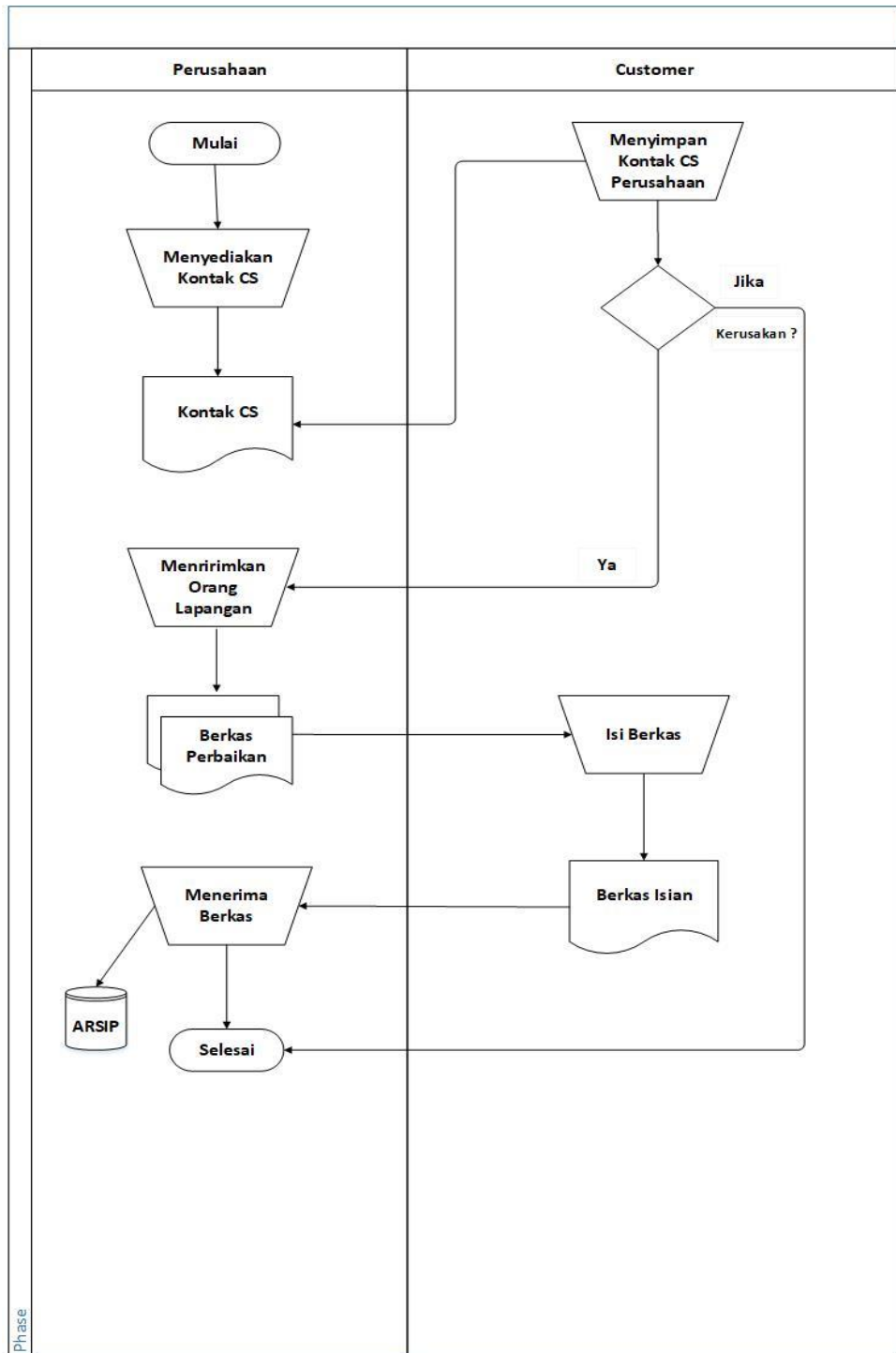
3.1.2. Analisis Sistem yang Sedang Berjalan

Analisis terhadap sistem yang sedang berjalan bertujuan untuk mengetahui bagaimana cara kerja suatu sistem dan mengetahui masalah yang dihadapi sistem untuk dapat dijadikan sebagai landasan ulasan perancangan sistem. Tahap analisis sistem perlu dilakukan dahulu sebelum dilakukan perancangan sistem. Tahap analisis ini merupakan tahapan yang penting didalam merancang suatu sistem. Analisis sistem yang dilakukan untuk mengetahui masalah apa yang sedang dihadapi oleh pihak perusahaan. Pada tahap perancangan ini, penulis menggambarkan sistem yang sedang berjalan dalam bentuk *flowmap*.

Berdasarkan hasil analisis yang dilakukan dengan metode wawancara, proses yang masih dilakukan sebagai berikut:

1. Pihak perusahaan memberikan kontak perusahaan kepada pihak *costumer* untuk menerima keluhan dari pihak *costumer*.
2. Pihak *costumer* tersebut akan menghubungi pihak perusahaan bila mana ada keluhan seputar perangkat penangkal petir.
3. Pihak perusahaan akan mengirimkan orang lapangan untuk memperbaiki keluhan yang di alami *costumer*.
4. Orang lapangan akan memberikan berkas perbaikan yang kemudian berkas tersebut harus diisi oleh *costumer*.
5. Setelah semuanya selesai, orang lapangan akan memberikan berkas tersebut kepada pihak perusahaan.

Ada pun penggambaran *flowmap* berdasarkan prosedur diatas digambarkan pada gambar 3.2 berikut:



GAMBAR: 3.2. Alur *Flowmap* yang sedang berjalan

3.1.3. Analisis Permasalahan

Analisis permasalahan merupakan dimana langkah-langkah ini diperlukan untuk mengetahui permasalahan apa saja yang terjadi di dalam sistem yang sedang berjalan. Adapun analisis permasalahan pada penelitian ini yaitu dalam berkomunikasi dengan *customer* kurang efisien karena dalam menyampaikan informasi kepada para *customer* baru maupun lama terkadang *customer service* lain yang berkomunikasi dengan *customer*, tidak sesuai dengan bagan struktur organisasi di perusahaan, dan juga hubungan *customer* tetap dengan perusahaan tidak bertahan lama karena respon yang di berikan kurang efisien jadi berpindah haluan mencari jasa penangkal petir lainnya.

3.1.4. Analisis Sistem yang Diusulkan

Berdasarkan analisis permasalahan yang dilakukan terhadap sistem yang sedang berjalan pada perusahaan ini, maka dari itu diusulkan membuat perancangan aplikasi untuk *customer service* berbasis *web* menggunakan *SMS gateway*. Maka dari itu dalam aplikasi yang diusulkan ini untuk meningkatkan hubungan baik antara perusahaan dengan *customer* dan memudahkan *customer service* dalam menyampaikan informasi kepada *customer* secara cepat dan juga efisien.

Dimana aplikasi tersebut *customer service* dapat melakukan semua operasi di dalam perangkat lunak ini seperti pengiriman pesan, pengiriman pesan tersiar, mengelola pesan terjadwal, mengelola kontak, mengelola

kontak grup, mengelola kontak masuk, mengelola kontak keluar dan mengelola kontak terkirim.

3.1.5. Analisis Perangkat yang digunakan

Agar aplikasi dapat dijalankan secara optimal, ada beberapa hal yang harus diperhatikan. Diantaranya adalah perangkat keras (*hardware*) dan perangkat lunak (*software*) harus sesuai spesifikasi yang direkomendasikan.

3.1.5.1. Perangkat Keras (*Hardware*)

Untuk membangun dan menjalankan aplikasi ini dibutuhkan persyaratan minimum perangkat keras agar aplikasi dapat berjalan tanpa kendala, tabel 3.1. menjelaskan persyaratan minimum perangkat keras (*hardware*) yang digunakan untuk membangun dan menjalankan aplikasi, dengan spesifikasi sebagai berikut :

TABEL: 3.1. Kebutuhan Minimum Perangkat Keras

	Perangkat	Developer	Pengguna
Perangkat keras (<i>Hardware</i>)	<i>Processor</i>	<i>Intel(R) Core(TM) 2 Duo CPU</i>	<i>Intel(R) Core(TM) 2 Duo CPU</i>
	<i>VGA</i>	<i>Mobile Intel(R) 965 Express Chipset Family</i>	<i>Mobile Intel(R) 965 Express Chipset Family</i>
	<i>Memori(RAM)</i>	4 GB	2 GB
	<i>Hardisk</i>	160 GB	80 GB
	<i>Monitor</i>	<i>Tabung/LCD/LED</i>	<i>Tabung/LCD/LED</i>
	<i>Modem</i>	<i>Wavecom M1306B</i>	<i>Wavecom M1306B</i>
	<i>Mouse</i>	<i>PS/2 atau USB</i>	<i>PS/2 atau USB</i>
	<i>Keyboard</i>	<i>PS/2 atau USB</i>	<i>PS/2 atau USB</i>

3.1.5.2. Perangkat Lunak (*Software*)

Selain perangkat keras untuk membangun dan menjalankan aplikasi ini dibutuhkan persyaratan minimum perangkat lunak (*software*) agar aplikasi dapat berjalan tanpa kendala, tabel 3.2. menjelaskan persyaratan minimum perangkat lunak yang digunakan. Dengan spesifikasi sebagai berikut :

TABEL: 3.2. Kebutuhan Minimum Perangkat Lunak

	Developer	Pengguna
Perangkat lunak (<i>Software</i>)	Sistem operasi Windows 7	Sistem operasi Windows 7
	<i>Web browser mozilla/crome/internet explorer</i>	<i>Web browser mozilla/crome/internet explorer</i>
	<i>Sublime Text build 3126</i>	XAMPP v3.2.2
	XAMPP v3.2.2	<i>MySQL v5.6.2.1</i>
	<i>MySQL v5.6.2.1</i>	<i>php MyAdmin v4.2.11</i>
	<i>php MyAdmin v4.2.11</i>	

3.2. Desain

Desain tahap ini dilakukan perancangan aplikasi yang akan di bangun, meliputi perancangan sistem dan perancangan basis data dan perancangan *interface* dari aplikasi yang akan di bangun.

3.2.1. Perancangan Sistem

Perancangan sistem ini bertujuan untuk mendeskripsikan interaksi satu atau lebih actor dengan sistem. Interaksi ini digambarkan dengan teknik UML (*Unified Modeling Language*).

3.2.1.1. Use Case Diagram

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

1. Deskripsi Aktor

Terdapat satu aktor dalam aplikasi ini, yaitu *customer service*, dimana *customer service* yang merupakan satu-satunya pengguna aplikasi ini yang langsung berkomunikasi dengan *customer*. *Customer service* dapat melakukan proses pengiriman pesan, pengiriman pesan tersiar, mengelola pesan terjadwal, mengelola kontak, mengelola kontak grup, mengelola kontak masuk, mengelola kontak keluar dan mengelola kontak terkirim.

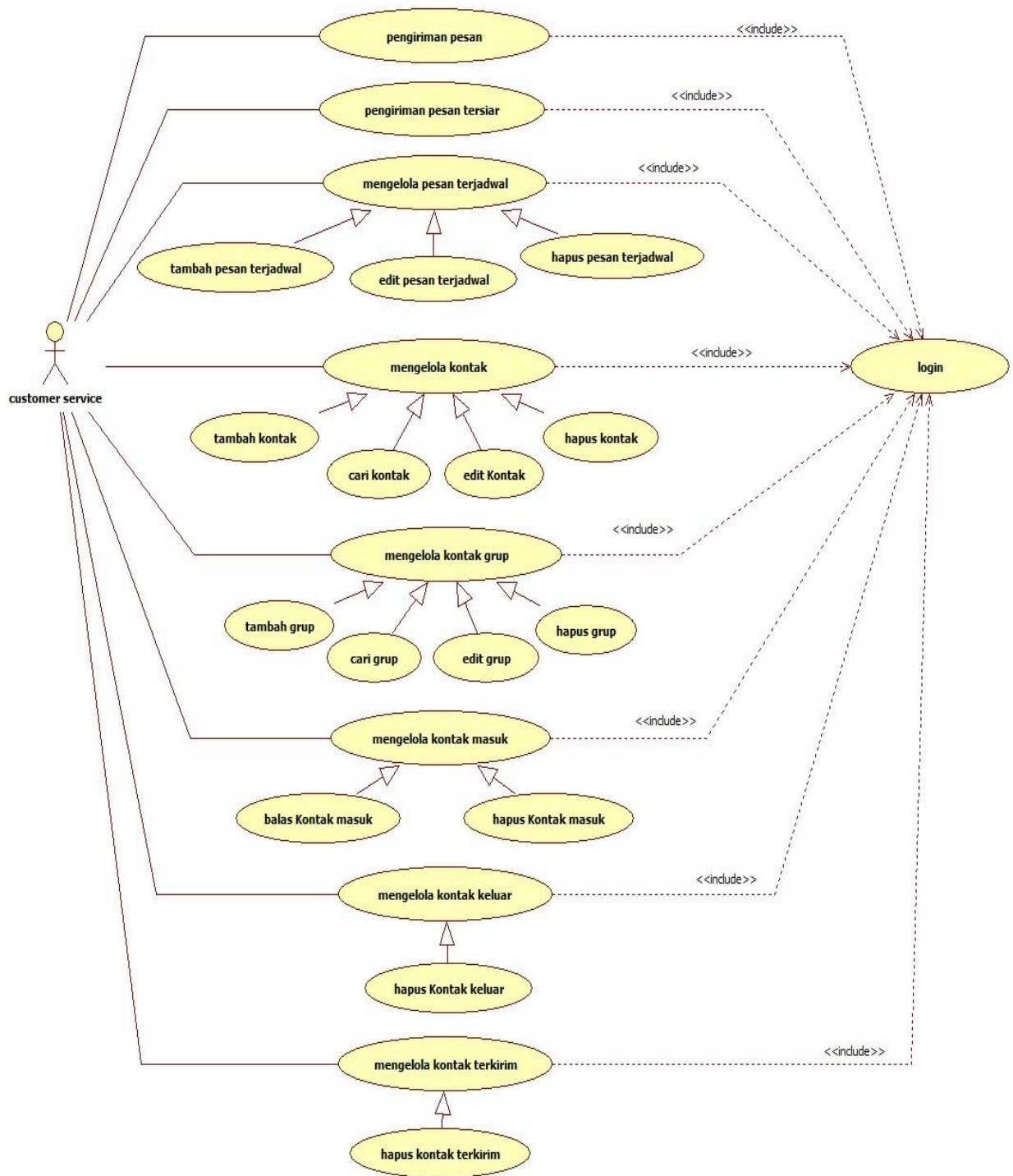
TABEL: 3.3. Deskripsi perancangan *Use Case Admin*

No	<i>Use Case</i>	Deskripsi
1	Login	Aktivitas ini dilakukan oleh <i>customer service</i> sebelum masuk ke <i>login</i> aplikasi dan <i>customer service</i> harus melakukan login terlebih dahulu.
2	pengiriman pesan	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk pengiriman pesan kepada <i>customer</i> dan terdapat kontak <i>customer</i> yang sudah tersimpan untuk berkomunikasi dengan <i>customer</i> .
3	pengiriman pesan tersiar	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk pengiriman pesan tersiar kepada semua kontak yang sudah disimpan. Pesan tersiar ini adalah pesan <i>broadcast SMS</i> atau <i>send all</i> , dimana <i>customer service</i> dapat memberikan informasi kepada semua kontak yang sudah tersimpan di <i>database</i> .

4	mengelola pesan terjadwal	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk mengelola pesan terjadwal kepada <i>customer</i> . Pesan terjadwal ini adalah pesan di tentukan oleh <i>customer service</i> kepada <i>customer</i> untuk memberikan informasi.
5	Tambah pesan terjadwal	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk membuat pesan terjadwal kepada <i>customer</i> . Pesan terjadwal ini adalah pesan di tentukan oleh <i>customer service</i> kepada <i>customer</i> untuk memberikan informasi.
6	Edit pesan terjadwal	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk mengedit pesan terjadwal yang terdapat pada <i>database</i> .
7	Hapus pesan terjadwal	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk menghapus pesan terjadwal yang terdapat pada <i>database</i> .
8	mengelola kontak	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk mengelola semua kontak. Dimana semua kontak akan di simpan menu kontak dan tersimpan di <i>database</i> .
9	tambah kontak	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk menambah kontak. Dimana kontak akan di simpan menu kontak dan tersimpan di <i>database</i> .
10	cari kontak	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk mencari kontak yang terdapat pada <i>database</i> .
11	Edit kontak	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk mengedit kontak yang terdapat pada <i>database</i> .
12	Hapus kontak	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk menghapus kontak yang terdapat pada <i>database</i> .
13	mengelola kontak grup	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk mengelola kontak grup. Setelah <i>customer service</i> menyimpan kontak di <i>database</i> maka akan di simpan kontak tersebut dalam semua grup yang sudah di buat.
14	Tambah grup	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk menambah kontak grup. Setelah <i>customer service</i> menyimpan kontak di <i>database</i> maka akan di simpan kontak tersebut dalam semua grup yang sudah di buat.

15	cari grup	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk mencari grup yang terdapat pada <i>database</i> .
16	Edit grup	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk mengedit kontak grup yang terdapat pada <i>database</i> .
17	Hapus grup	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk menghapus kontak grup yang terdapat pada <i>database</i> .
18	mengelola kontak masuk	Merupakan fitur ketika ada pesan SMS yang masuk yang tersimpan di <i>database</i> .
19	balas kontak masuk	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk membalas sebuah pesan.
20	Hapus kontak masuk	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk menghapus kontak masuk yang terdapat pada <i>database</i> .
21	mengelola kontak keluar	Merupakan fitur ketika ada pesan SMS sedang dikirim yang akan tersimpan di <i>database</i> .
22	Hapus kontak keluar	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk menghapus kontak keluar yang terdapat pada <i>database</i> .
23	Mengelola kontak terkirim	Merupakan fitur ketika SMS berhasil dikirim yang akan tersimpan di <i>database</i> .
24	Hapus kontak terkirim	Merupakan fitur yang dapat dilakukan oleh <i>customer service</i> untuk menghapus kontak terkirim yang terdapat pada <i>database</i> .

Adapun *Use Case diagram* untuk perancangan aplikasi untuk *customer service* berbasis *web* menggunakan *SMS gateway* digambarkan pada gambar 3.3.



GAMBAR: 3.3. Use Case Diagram aplikasi untuk customer service

Dari Gambar 3.3. terlihat *customer service* dapat melakukan pengiriman pesan, pengiriman pesan tersiar, mengelola pesan terjadwal, mengelola kontak, mengelola kontak grup, mengelola kontak masuk, mengelola kontak keluar serta mengelola kontak terkirim.

Berikut adalah proses skenario proses yang terdapat pada *use case* Berdasarkan *use case diagram* aplikasi diatas maka dibuatlah skenario dari tiap proses yang ada pada *use case diagram* tersebut untuk memudahkan dalam menganalisa skenario yang akan kita gunakan dalam fase selanjutnya. Untuk memudahkan proses analisa, maka skenario dari setiap *use case* akan dibuat menjadi beberapa tabel skenario sebagai berikut :

TABEL: 3.4. Skenario *Use Case login*

Skenario use case	
Nama Use Case	<i>Login</i>
Aktor	<i>customer service</i>
Tujuan	Mengakses fungsi menu utama
Aksi Aktor	Respon Sistem
Kondisi awal	<i>Form login</i>
1. <i>Input username dan password</i> lalu menekan tombol <i>login</i> .	
	2. Validasi <i>username</i> dan <i>password</i> sesuai dengan hak akses pada <i>database</i> .
	3. Menampilkan <i>dashboard</i> sesuai hak akses jika benar, jika salah kembali pada tampilan <i>login</i> .
Kondisi akhir	Menampilkan menu utama

TABEL: 3.5. Skenario *Use Case* pengiriman pesan

Skenario use case	
Nama	pengiriman pesan
Aktor	<i>customer service</i>
Tujuan	Untuk mengirim pesan
Aksi Aktor	Respon Sistem
Kondisi awal	menu utama
1. pilih SMS.	
	2. menampilkan pilihan SMS.
3. <i>customer service</i> memilih tulis pesan.	
	4. menampilkan modal <i>form</i> tulis pesan.
5. <i>input</i> kontak yang sudah terdaftar dan isi pesan lalu klik kirim.	
	6. sistem menampilkan pesan berhasil terkirim.
Kondisi akhir	Menampilkan menu utama

TABEL: 3.6. Skenario *Use Case* pengiriman pesan tersiar

Skenario use case	
Nama	Pengiriman pesan tersiar
Aktor	<i>customer service</i>
Tujuan	Untuk mengirim pesan tersiar
Aksi Aktor	Respon Sistem
Kondisi awal	menu utama
1. pilih SMS.	
	2. menampilkan pilihan SMS.
3. <i>customer service</i> memilih pesan tersiar.	
	4. menampilkan modal <i>form</i> pesan tersiar.
5. masukan kontak grup yang sudah terdaftar dan isi pesan lalu klik kirim.	
	6. sistem menampilkan pesan berhasil terkirim.
Kondisi akhir	Menampilkan menu utama

TABEL: 3.7. Skenario *Use Case* mengelola pesan terjadwal

Skenario use case	
Nama	Mengelola pesan terjadwal
Aktor	<i>customer service</i>
Tujuan	Untuk menampilkan menu pesan terjadwal
Aksi Aktor	Respon Sistem
Kondisi awal	menu utama
1.pilih auto SMS.	
	2. menampilkan pilihan auto SMS.
3. <i>customer service</i> memilih pesan terjadwal.	
	4. menampilkan menu pesan terjadwal (tambah pesan terjadwal, edit pesan terjadwal, Hapus pesan terjadwal).
Kondisi akhir	Menampilkan menu pesan terjadwal

TABEL: 3.8. Skenario *Use Case* tambah pesan terjadwal

Skenario use case	
Nama	tambah pesan terjadwal
Aktor	<i>customer service</i>
Tujuan	menambah pesan terjadwal
Aksi Aktor	Respon Sistem
Kondisi awal	menu pesan terjadwal
1. memilih tambah pesan terjadwal.	
	2. menampilkan modal <i>form</i> tambah pesan terjadwal.
3. <i>input</i> kontak, isi pesan dan tanggal yang akan dijadwalkan untuk di kirim, kemudian klik simpan.	
	4. pesan terjadwal sudah buat dan menampilkan menu pesan terjadwal.
Kondisi akhir	Menampilkan menu pesan terjadwal

TABEL: 3.9. Skenario *Use Case* edit pesan terjadwal

Skenario use case	
Nama	edit pesan terjadwal
Aktor	<i>customer service</i>
Tujuan	Mengedit pesan terjadwal
Aksi Aktor	Respon Sistem
Kondisi awal	menu pesan terjadwal
1. memilih pesan terjadwal yang akan di edit.	
	2. menampilkan modal <i>form</i> edit pesan terjadwal.
3. edit kontak, isi pesan dan tanggal yang akan dijadwalkan untuk di kirim, kemudian klik simpan.	
	4. perubahan sudah di ubah dan menampilkan menu pesan terjadwal.
Kondisi akhir	Menampilkan menu pesan terjadwal.

TABEL: 3.10. Skenario *Use Case* hapus pesan terjadwal

Skenario use case	
Nama	Hapus pesan terjadwal
Aktor	<i>customer service</i>
Tujuan	Menghapus pesan terjadwal
Aksi Aktor	Respon Sistem
Kondisi awal	Menu pesan terjadwal
1. memilih pesan terjadwal yang akan di hapus.	
	2. menghapus pesan terjadwal yang dipilih <i>customer service</i> dari <i>database</i> .
	3. menampilkan menu pesan terjadwal.
Kondisi akhir	Menampilkan menu pesan terjadwal.

TABEL: 3.11. Skenario *Use Case* mengelola kontak

Skenario use case	
Nama	Mengelola kontak
Aktor	<i>customer service</i>
Tujuan	Untuk menampilkan menu daftar kontak
Aksi Aktor	Respon Sistem
Kondisi awal	Menu utama
1. pilih kontak.	
	2. menampilkan pilihan kontak.
3. <i>customer service</i> memilih daftar kontak.	
	4. menampilkan menu daftar kontak (cari kontak, tambah kontak, edit kontak, hapus kontak).
	5. menampilkan menu daftar kontak.
Kondisi akhir	menampilkan menu daftar kontak

TABEL: 3.12. Skenario *Use Case* tambah kontak

Skenario use case	
Nama	tambah kontak
Aktor	<i>customer service</i>
Tujuan	menambah kontak
Aksi Aktor	Respon Sistem
Kondisi awal	menu daftar kontak
1.pilih tambah kontak.	
	2. menampilkan modal <i>form</i> tambah kontak.
3. <i>input</i> nama, no telepon, tanggal lahir, email dan foto, lalu klik tambah.	
	4. menampilkan kontak berhasil di simpan.
Kondisi akhir	menampilkan menu daftar kontak

TABEL: 3.13. Skenario *Use Case* cari kontak

Skenario use case	
Nama	Cari kontak
Aktor	<i>customer service</i>
Tujuan	Mencari kontak
Aksi Aktor	Respon Sistem
Kondisi awal	menu daftar kontak
1.mecari nama kontak.	
	2. menampilkan kontak yang dicari.
Kondisi akhir	menampilkan menu daftar kontak

TABEL: 3.14. Skenario *Use Case* edit kontak

Skenario use case	
Nama	Edit Kontak
Aktor	<i>customer service</i>
Tujuan	Mengedit kontak
Aksi Aktor	Respon Sistem
Kondisi awal	menu daftar kontak
1. memilih kontak yang akan di edit.	
	2. menampilkan modal <i>form</i> edit kontak.
3. mengedit kontak, kemudian klik edit.	
	4. menampilkan kontak berhasil di edit.
Kondisi akhir	menampilkan menu daftar kontak

TABEL: 3.15. Skenario *Use Case* hapus kontak

Skenario use case	
Nama	Hapus Kontak
Aktor	<i>customer service</i>
Tujuan	Menghapus kontak
Aksi Aktor	Respon Sistem
Kondisi awal	menu daftar kontak
1. memilih kontak yang akan di hapus.	
	2. menghapus kontak yang dipilih <i>customer service</i> dari <i>database</i> .
	3. menampilkan berhasil menghapus kontak.
Kondisi akhir	menampilkan menu daftar kontak

TABEL: 3.16. Skenario *Use Case* mengelola kontak grup

Skenario use case	
Nama	Mengelola kontak grup
Aktor	<i>customer service</i>
Tujuan	Untuk menampilkan menu daftar grup
Aksi Aktor	Respon Sistem
Kondisi awal	Halaman utama
1. pilih kontak.	
	2. menampilkan pilihan kontak.
3. <i>customer service</i> memilih daftar grup.	
	4. menampilkan <i>form</i> daftar grup (tambah grup, cari grup, edit grup, hapus grup).
	5. menampilkan menu daftar grup.
Kondisi akhir	menampilkan menu daftar grup

TABEL: 3.17. Skenario *Use Case* tambah grup

Skenario use case	
Nama	tambah grup
Aktor	<i>customer service</i>
Tujuan	menambah grup
Aksi Aktor	Respon Sistem
Kondisi awal	menu daftar grup
1. <i>customer service</i> memilih tambah grup dan klik.	
	2. menampilkan modal <i>form</i> grup.
3. <i>input</i> nama grup, kemudian klik simpan.	
	4. menampilkan menu daftar grup.
Kondisi akhir	menampilkan menu daftar grup

TABEL: 3.18. Skenario *Use Case* cari grup

Skenario use case	
Nama	Cari grup
Aktor	<i>customer service</i>
Tujuan	Mencari kontak grup
Aksi Aktor	Respon Sistem
Kondisi awal	menu daftar grup
1. mencari nama grup.	
	2. menampilkan grup yang dicari.
Kondisi akhir	menampilkan menu daftar grup

TABEL: 3.19. Skenario *Use Case* edit grup

Skenario use case	
Nama	Edit grup
Aktor	<i>customer service</i>
Tujuan	Mengedit kontak grup
Aksi Aktor	Respon Sistem
Kondisi awal	menu daftar grup
1. memilih kontak grup yang akan di edit.	
	2. menampilkan modal <i>form</i> edit grup.
3. edit nama kontak grup, kemudian klik simpan.	
	4. menampilkan menu daftar grup.
Kondisi akhir	menampilkan menu daftar grup

TABEL: 3.20. Skenario *Use Case* hapus grup

Skenario use case	
Nama	hapus grup
Aktor	<i>customer service</i>
Tujuan	Menghapus kontak grup
Aksi Aktor	Respon Sistem
Kondisi awal	menu daftar grup
1. memilih grup yang akan di hapus.	
	2. menghapus kontak grup yang dipilih <i>customer service</i> dari <i>database</i> .
	3. menampilkan menu daftar grup.
Kondisi akhir	menampilkan menu daftar grup

TABEL: 3.21. Skenario *Use Case* mengelola kontak masuk

Skenario use case	
Nama	Mengelola kontak masuk
Aktor	<i>customer service</i>
Tujuan	Menampilkan menu kontak masuk
Aksi Aktor	Respon Sistem
Kondisi awal	Menu utama
1. pilih SMS.	
	2. menampilkan pilihan SMS.
3. <i>customer service</i> memilih kontak masuk.	
	4. menampilkan menu kontak masuk (balas kontak masuk, hapus kontak masuk).
	5. menampilkan menu kontak masuk.
Kondisi akhir	menampilkan menu kontak masuk

TABEL: 3.22. Skenario *Use Case* balas kontak masuk

Skenario use case	
Nama	Balas kontak masuk
Aktor	<i>customer service</i>
Tujuan	Membalas pesan kontak masuk
Aksi Aktor	Respon Sistem
Kondisi awal	menu kontak masuk
1. <i>customer service</i> memilih pesan yang akan di balas.	
	2. menampilkan modal <i>form</i> pesan.
3. <i>input</i> pesan kemudian klik kirim.	
	4. menampilkan pesan berhasil dikirim.
Kondisi akhir	menampilkan menu kontak masuk

TABEL: 3.23. Skenario *Use Case* hapus kontak masuk

Skenario use case	
Nama	hapus kontak masuk
Aktor	<i>customer service</i>
Tujuan	menghapus pesan kontak masuk
Aksi Aktor	Respon Sistem
Kondisi awal	menu kontak masuk
1. <i>customer service</i> memilih pesan yang akan dihapus.	
	2. menghapus pesan yang dipilih <i>customer service</i> dari <i>database</i> .
	3. menampilkan menu kontak masuk.
Kondisi akhir	menampilkan menu kontak masuk

TABEL: 3.24. Skenario *Use Case* mengelola kontak keluar

Skenario use case	
Nama	Mengelola kontak keluar
Aktor	<i>customer service</i>
Tujuan	Melihat status pengiriman
Aksi Aktor	Respon Sistem
Kondisi awal	Menu utama
1. pilih SMS.	
	2. menampilkan pilihan SMS.
3. <i>customer service</i> memilih kontak keluar.	
	4. menampilkan menu kontak keluar.
Kondisi akhir	menampilkan menu kontak keluar

TABEL: 3.25. Skenario *Use Case* hapus kontak keluar

Skenario use case	
Nama	hapus kontak keluar
Aktor	<i>customer service</i>
Tujuan	menghapus pesan kontak keluar
Aksi Aktor	Respon Sistem
Kondisi awal	Menu Kontak keluar
1. memilih pesan yang akan dihapus.	
	2. menghapus pesan yang dipilih admin dari <i>database</i> .
	3. menampilkan menu kontak keluar.
Kondisi akhir	menampilkan menu kontak keluar

TABEL: 3.26. Skenario *Use Case* mengelola kontak terkirim

Skenario use case	
Nama	Mengelola kontak terkirim
Aktor	<i>customer service</i>
Tujuan	Melihat pesan terkirim
Aksi Aktor	Respon Sistem
Kondisi awal	Menu utama
1. pilih SMS.	
	2. menampilkan pilihan SMS.
3. <i>customer service</i> memilih kontak terkirim.	
	4. menampilkan menu kontak terkirim.
Kondisi akhir	menampilkan menu kontak terkirim

TABEL: 3.27. Skenario *Use Case* hapus kontak terkirim

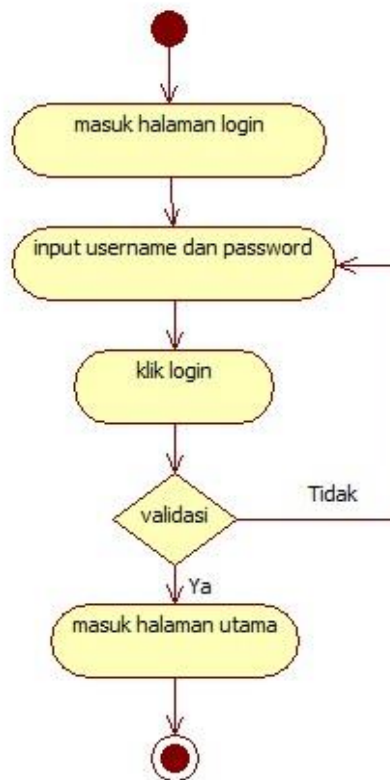
Skenario use case	
Nama	hapus kontak terkirim
Aktor	<i>customer service</i>
Tujuan	menghapus pesan kontak terkirim
Aksi Aktor	Respon Sistem
Kondisi awal	Menu Kontak terkirim
1. memilih pesan yang yang akan dihapus.	
	2. menghapus pesan yang dipilih admin dari <i>database</i> .
	3. menampilkan menu kontak terkirim.
Kondisi akhir	menampilkan menu kontak terkirim

3.2.1.2. Activity Diagram

Activity diagram adalah diagram yang menunjukkan aktivitas dari setiap fungsi yang ada, biasanya menggambarkan *workflow* atau proses bisnis, bisa juga menggambarkan aktivitas menu yang ada pada aplikasi. Berikut *activity diagram* perancangan aplikasi untuk *customer service* berbasis *web* menggunakan *SMS Gateway* sebagai berikut:

1. Activity Diagram Login

Activity diagram pada gambar 3.4. menggambarkan proses login. Di sini dijelaskan agar dapat masuk ke halaman utama *customer service* harus melakukan login terlebih dahulu, tentunya ketika melakukan login harus sudah memiliki *username* dan *password*. Saat *customer service* memasukan *username* dan *password* maka sistem akan melakukan validasi data dari *username* dan *password* tersebut, dan jika sesuai maka akan masuk ke halaman utama, jika tidak sesuai akan muncul pesan “gagal login”.



GAMBAR: 3.4. Activity Diagram Login

2. *Activity Diagram* pengiriman pesan

Activity diagram pada gambar 3.5. menggambarkan proses dalam pengiriman pesan. Disini dijelaskan agar dapat masuk ke halaman pengiriman pesan *customer service* harus memilih menu SMS kemudian memilih tulis pesan dan menampilkan modal *form* tulis pesan, *customer service* menginput kontak dan isi pesan yang akan di kirim, kemudian klik kirim.



GAMBAR: 3.5. *Activity Diagram* pengiriman pesan

3. *Activity Diagram* pengiriman pesan tersiar

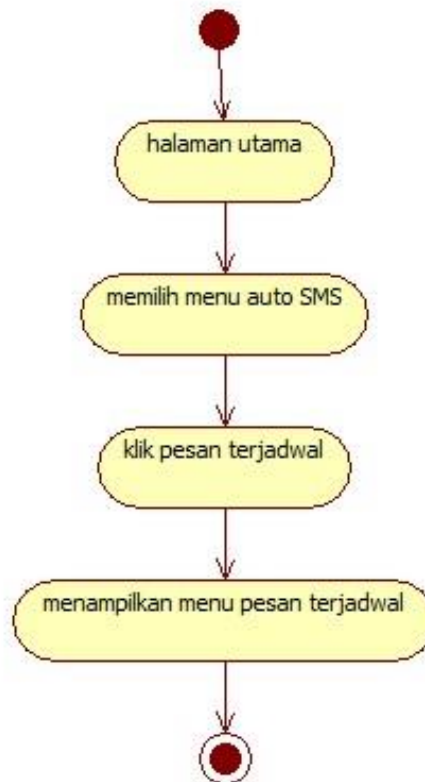
Activity diagram pada gambar 3.6. menggambarkan proses dalam pengiriman pesan tersiar atau *broadcast*. Disini dijelaskan agar dapat masuk ke halaman pengiriman pesan *customer service* harus memilih menu SMS kemudian memilih pesan tersiar dan menampilkan modal *form* pesan tersiar, *customer service* menginput kontak grup dan isi pesan yang akan di kirim, kemudian klik kirim.



GAMBAR: 3.6. *Activity Diagram* pengiriman pesan tersiar

4. *Activity Diagram* Mengelola pesan terjadwal

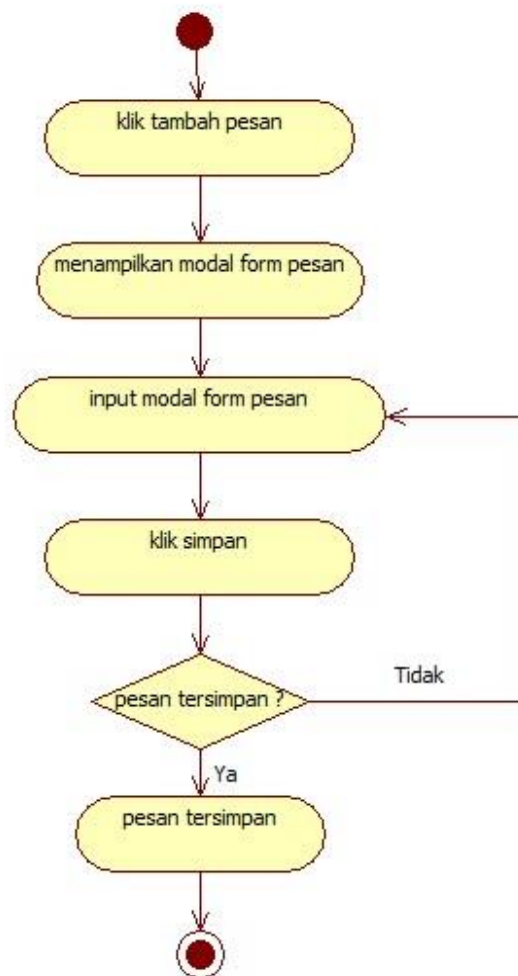
Activity diagram pada gambar 3.7. menggambarkan proses dalam mengelola pesan terjadwal. Disini dijelaskan *customer service* agar dapat masuk ke menu mengelola pesan terjadwal memilih menu auto SMS dan klik pesan terjadwal kemudian akan menampilkan mengelola pesan terjadwal.



GAMBAR: 3.7. *Activity Diagram* Mengelola pesan terjadwal

5. Activity Diagram Tambah pesan terjadwal

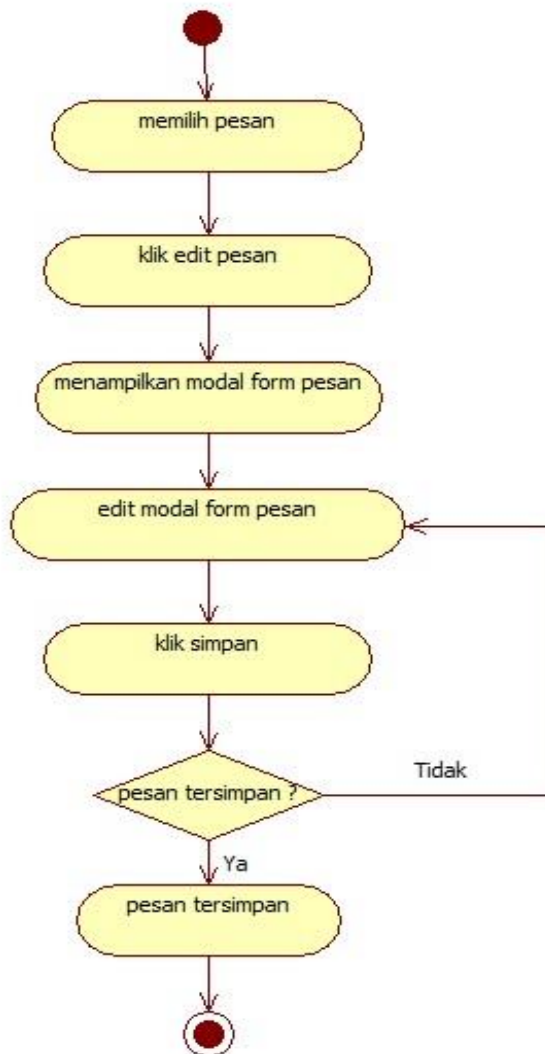
Activity diagram pada gambar 3.8. menggambarkan proses dalam menambah pesan terjadwal. Disini dijelaskan agar dapat membuat pesan terjadwal, *customer service* memilih tambah pesan kemudian menampilkan modal form pesan terjadwal, *customer service* menginput modal form pesan, terdapat *decision* jika tidak mengisi pesan sistem menampilkan “pesan harus di isi” jika pesan sudah di *input* kemudian klik simpan maka pesan terjadwal akan tersimpan.



GAMBAR: 3.8. Activity Diagram tambah pesan terjadwal

6. *Activity Diagram* edit pesan terjadwal

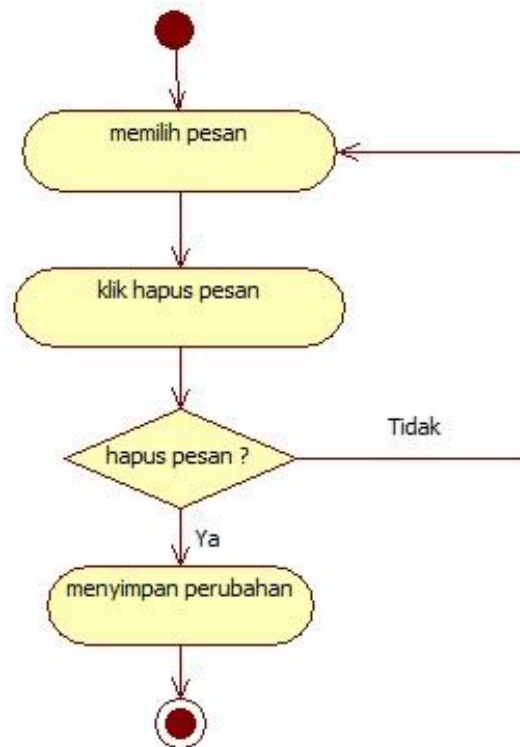
Activity diagram pada gambar 3.9. menggambarkan proses dalam mengedit pesan terjadwal. Disini dijelaskan agar dapat mengubah pesan terjadwal, *customer service* memilih pesan kemudian klik edit pesan lalu akan menampilkan modal *form* pesan, *customer service* mengedit modal *form* pesan kemudian klik simpan dan pesan terjadwal akan tersimpan.



GAMBAR: 3.9. *Activity Diagram* edit pesan terjadwal

7. Activity Diagram hapus pesan terjadwal

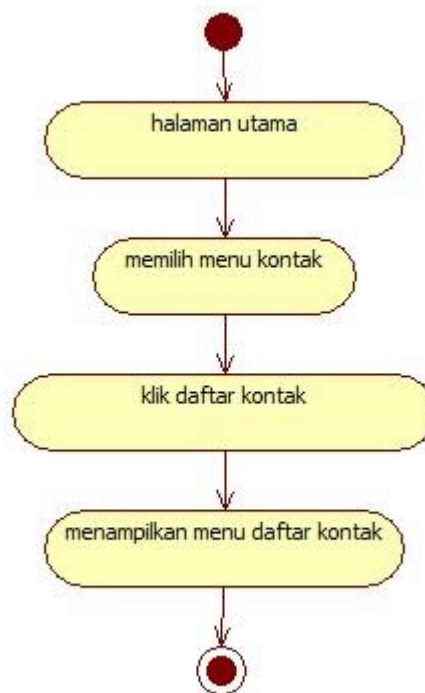
Activity diagram pada gambar 3.10. menggambarkan proses dalam menghapus pesan terjadwal. Disini dijelaskan agar dapat menghapus pesan terjadwal, *customer service* memilih pesan yang akan di hapus lalu klik hapus pesan, sistem menampilkan apakah pesan di hapus atau tidak, jika “tidak” akan kembali ke pilihan pesan dan jika “iya” akan menghapus pesan.



GAMBAR: 3.10. Activity Diagram hapus pesan terjadwal

8. *Activity Diagram* Mengelola kontak

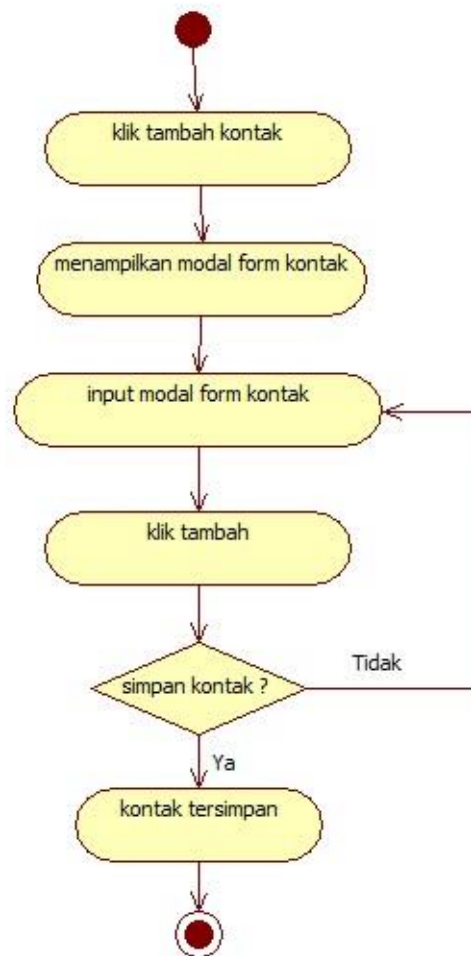
Activity diagram pada gambar 3.11. menggambarkan proses dalam mengelola kontak. Disini dijelaskan *customer service* agar dapat masuk ke menu mengelola kontak, ketika memilih menu kontak dan klik daftar kontak kemudian akan menampilkan mengelola kontak.



GAMBAR: 3.11. *Activity Diagram* Mengelola kontak

9. Activity Diagram tambah kontak

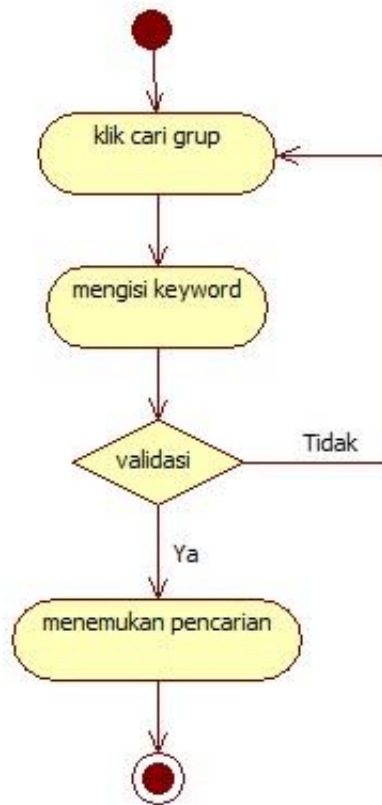
Activity diagram pada gambar 3.12. menggambarkan proses dalam menambah kontak. Disini dijelaskan agar dapat menambah kontak, *customer service* memilih tambah kontak kemudian menampilkan modal form kontak, *customer service* menginput modal form kontak, terdapat *decision* jika tidak menginput sistem menampilkan “harap di isi” jika sudah menginput kemudian klik tambah maka kontak akan tersimpan.



GAMBAR: 3.12. Activity Diagram tambah kontak

10. *Activity Diagram* cari kontak

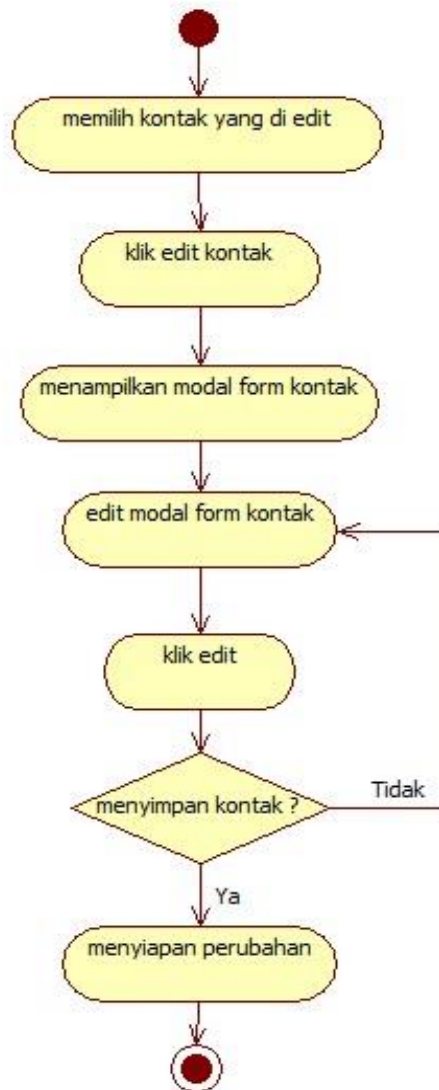
Activity diagram pada gambar 3.13. menggambarkan proses dalam mencari sebuah kontak. Disini dijelaskan agar dapat mencari kontak dengan cara *customer service* mengisi *keyword*.



GAMBAR: 3.13. *Activity Diagram* cari kontak

11. Activity Diagram edit kontak

Activity diagram pada gambar 3.14. menggambarkan proses dalam mengedit kontak. Disini dijelaskan agar dapat mengedit kontak, *customer service* memilih kontak yang akan di edit lalu akan menampilkan modal *form* kontak, *customer service* mengedit modal *form* kontak, kemudian klik edit dan kontak akan tersimpan.



GAMBAR: 3.14. Activity Diagram edit kontak

12. Activity Diagram hapus kontak

Activity diagram pada gambar 3.15. menggambarkan proses dalam menghapus kontak. Disini dijelaskan agar dapat menghapus kontak, *customer service* memilih kontak yang akan di hapus lalu klik hapus kontak, sistem menampilkan apakah pesan di hapus atau tidak, jika “tidak” akan kembali ke pilihan pesan dan jika “iya” akan menghapus kontak.



GAMBAR: 3.15. Activity Diagram hapus kontak

13. Activity Diagram Mengelola kontak grup

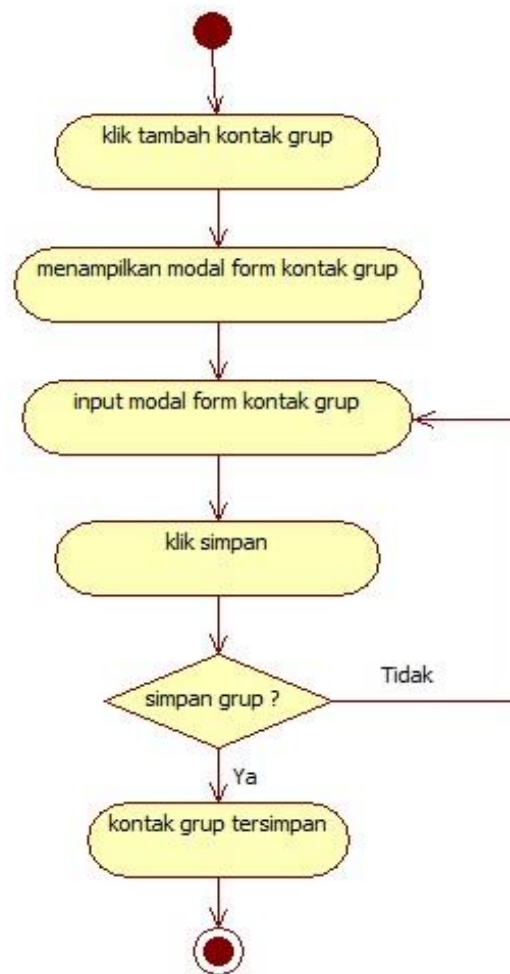
Activity diagram pada gambar 3.16. menggambarkan proses dalam mengelola kontak grup. Disini dijelaskan *customer service* agar dapat masuk ke menu mengelola kontak grup, ketika memilih menu kontak dan klik daftar grup kemudian akan menampilkan mengelola kontak grup.



GAMBAR: 3.16. Activity Diagram Mengelola kontak grup

14. Activity Diagram tambah grup

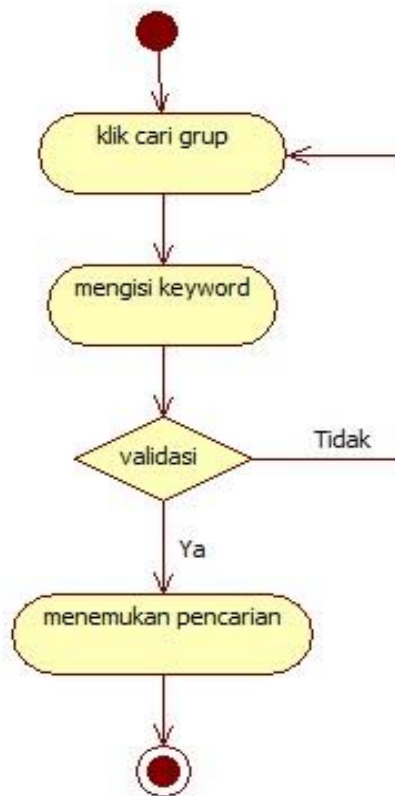
Activity diagram pada gambar 3.17. menggambarkan proses dalam menambah grup. Disini dijelaskan agar dapat menambah kontak grup, *customer service* memilih tambah kontak grup kemudian menampilkan modal form kontak grup, *customer service* menginput modal form kontak grup, terdapat *decision* jika tidak menginput sistem menampilkan “harap di isi” jika sudah menginput kemudian klik simpan maka kontak grup akan tersimpan.



GAMBAR: 3.17. Activity Diagram tambah grup

15. *Activity Diagram* cari grup

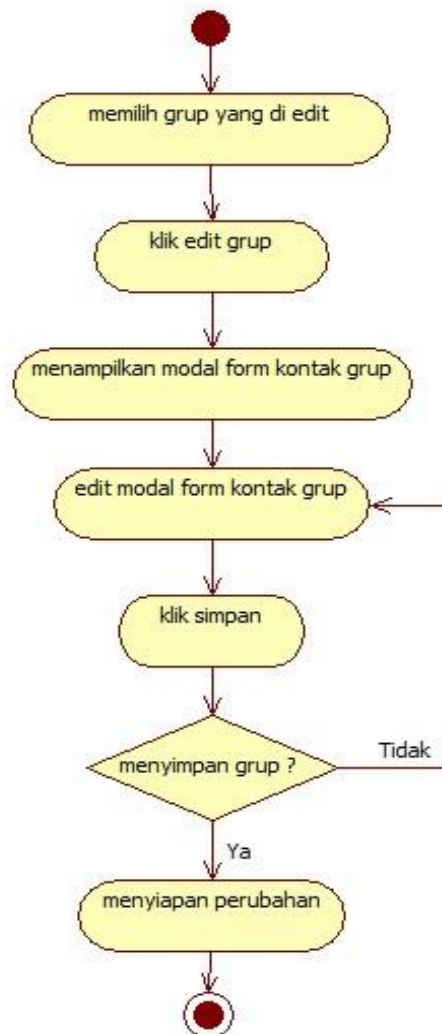
Activity diagram pada gambar 3.18. menggambarkan proses dalam mencari grup. Disini dijelaskan agar dapat mencari kontak grup dengan cara *customer service* mengisi *keyword*.



GAMBAR: 3.18. *Activity Diagram* cari grup

16. Activity Diagram edit grup

Activity diagram pada gambar 3.19. menggambarkan proses dalam edit grup. Disini dijelaskan agar dapat mengedit kontak grup, *customer service* memilih grup yang akan di edit lalu akan menampilkan modal form kontak grup, *customer service* mengedit modal form kontak grup, kemudian klik edit dan kontak grup akan tersimpan.



GAMBAR: 3.19. Activity Diagram edit grup

17. Activity Diagram hapus grup

Activity diagram pada gambar 3.20. menggambarkan proses dalam hapus grup. Disini dijelaskan agar dapat menghapus kontak grup, *customer service* memilih kontak grup yang akan di hapus lalu klik hapus grup, sistem menampilkan apakah pesan di hapus atau tidak, jika “tidak” akan kembali ke pilihan pesan dan jika “iya” akan menghapus kontak grup.



GAMBAR: 3.20. Activity Diagram hapus grup

18. Activity Diagram Mengelola kontak masuk

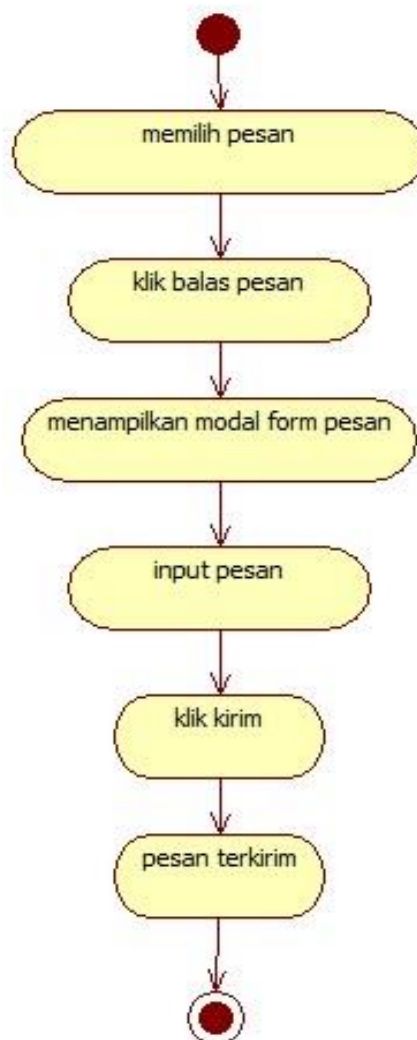
Activity diagram pada gambar 3.21. menggambarkan proses dalam mengelola kontak masuk. Disini dijelaskan *customer service* agar dapat masuk ke menu mengelola kontak masuk, ketika memilih menu SMS dan klik kontak masuk kemudian akan menampilkan mengelola kontak masuk.



GAMBAR: 3.21. Activity Diagram Mengelola kontak masuk

19. Activity Diagram balas kontak masuk

Activity diagram pada gambar 3.22. menggambarkan proses dalam membalas kontak masuk. Disini dijelaskan agar dapat membalas kontak, *customer service* memilih pesan kemudian klik balas pesan menampilkan modal form pesan, *customer service* menginput pesan lalu klik kirim dan pesan pun telah terkirim.



GAMBAR: 3.22. Activity Diagram balas kontak masuk

20. *Activity Diagram* hapus kontak masuk

Activity diagram pada gambar 3.23. menggambarkan proses dalam menghapus kontak masuk. Disini dijelaskan agar dapat menghapus kontak pesan, *customer service* memilih pesan masuk yang akan di hapus lalu klik hapus, sistem menampilkan apakah pesan di hapus atau tidak, jika “tidak” akan kembali ke pilihan pesan dan jika “iya” akan menghapus pesan masuk.



GAMBAR: 3.23. *Activity Diagram* hapus kontak masuk

21. Activity Diagram Mengelola kontak keluar

Activity diagram pada gambar 3.24. menggambarkan proses dalam mengelola kontak keluar. Disini dijelaskan *customer service* agar dapat masuk ke menu mengelola kontak keluar, ketika memilih menu SMS dan klik kontak keluar kemudian akan menampilkan mengelola kontak keluar.



GAMBAR: 3.24. Activity Diagram Mengelola kontak keluar

22. Activity Diagram hapus kontak keluar

Activity diagram pada gambar 3.25. menggambarkan proses dalam menghapus kontak keluar. Disini dijelaskan agar dapat menghapus kontak keluar, *customer service* memilih pesan keluar yang akan di hapus lalu klik hapus, sistem menampilkan apakah pesan di hapus atau tidak, jika “tidak” akan kembali ke pilihan pesan dan jika “iya” akan menghapus pesan keluar.



GAMBAR: 3.25. Activity Diagram hapus kontak keluar

23. Activity Diagram Mengelola kontak terkirim

Activity diagram pada gambar 3.26. menggambarkan proses dalam mengelola kontak terkirim. Disini dijelaskan *customer service* agar dapat masuk ke menu mengelola kontak terkirim, ketika memilih menu SMS dan klik kontak terkirim kemudian akan menampilkan mengelola kontak terkirim.



GAMBAR: 3.26. Activity Diagram Mengelola kontak terkirim

24. Activity Diagram hapus kontak terkirim

Activity diagram pada gambar 3.27. menggambarkan proses dalam menghapus kontak keluar. Disini dijelaskan agar dapat menghapus kontak terkirim, *customer service* memilih pesan terkirim yang akan di hapus lalu klik hapus, sistem menampilkan apakah pesan di hapus atau tidak, jika “tidak” akan kembali ke pilihan pesan dan jika “iya” akan menghapus pesan keluar.



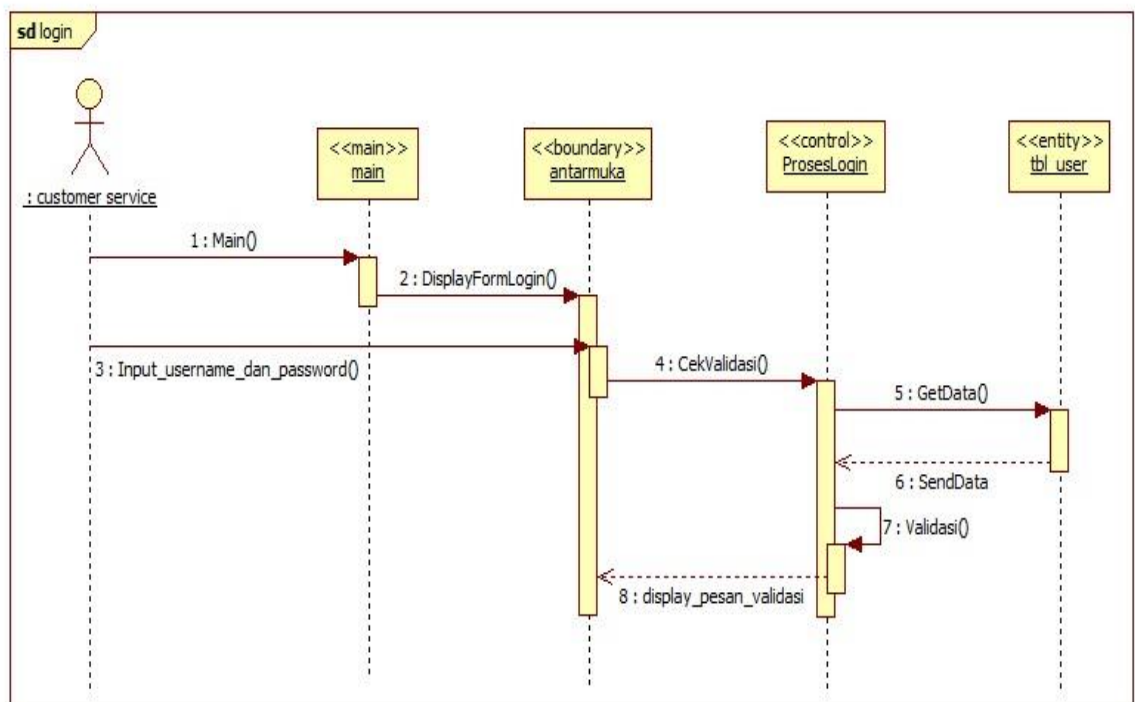
GAMBAR: 3.27. Activity Diagram hapus kontak terkirim

3.2.1.3. Sequence Diagram

Sequence Diagram adalah suatu diagram yang menggambarkan interaksi antara sebuah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem. *Sequence Diagram* perancangan aplikasi untuk *customer service* berbasis *web* menggunakan *SMS Gateway* sebagai berikut:

1. Sequence Diagram Login

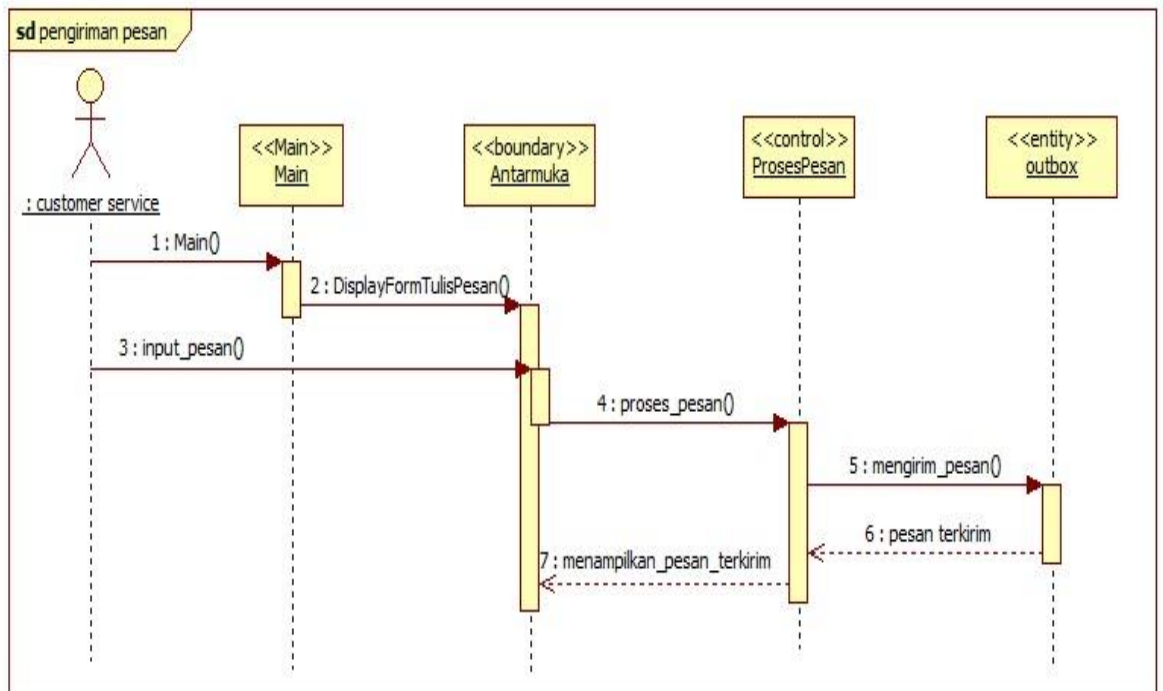
Pada gambar 3.28. menunjukkan *sequence diagram login*, proses *customer service* untuk masuk ke halaman utama.



GAMBAR: 3.28. *Sequence Diagram login*

2. Sequence Diagram pengiriman pesan

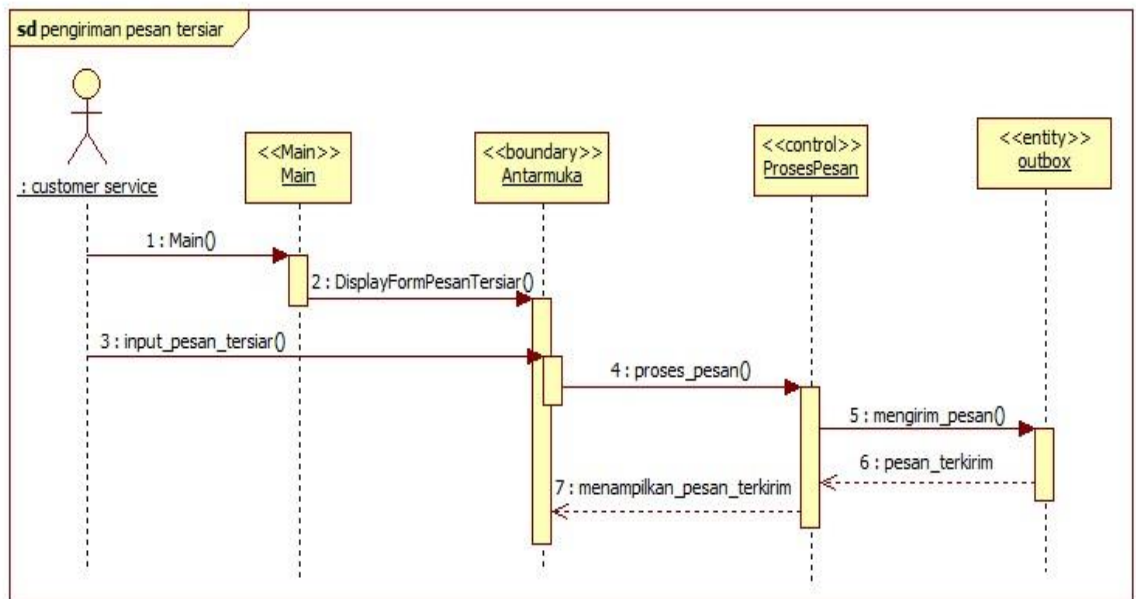
Pada gambar 3.29. menunjukkan *sequence diagram* pengiriman pesan, dimana proses *customer service* dalam mengirim sebuah pesan.



GAMBAR: 3.29. *Sequence Diagram* pengiriman pesan

3. *Sequence Diagram* pengiriman pesan tersiar

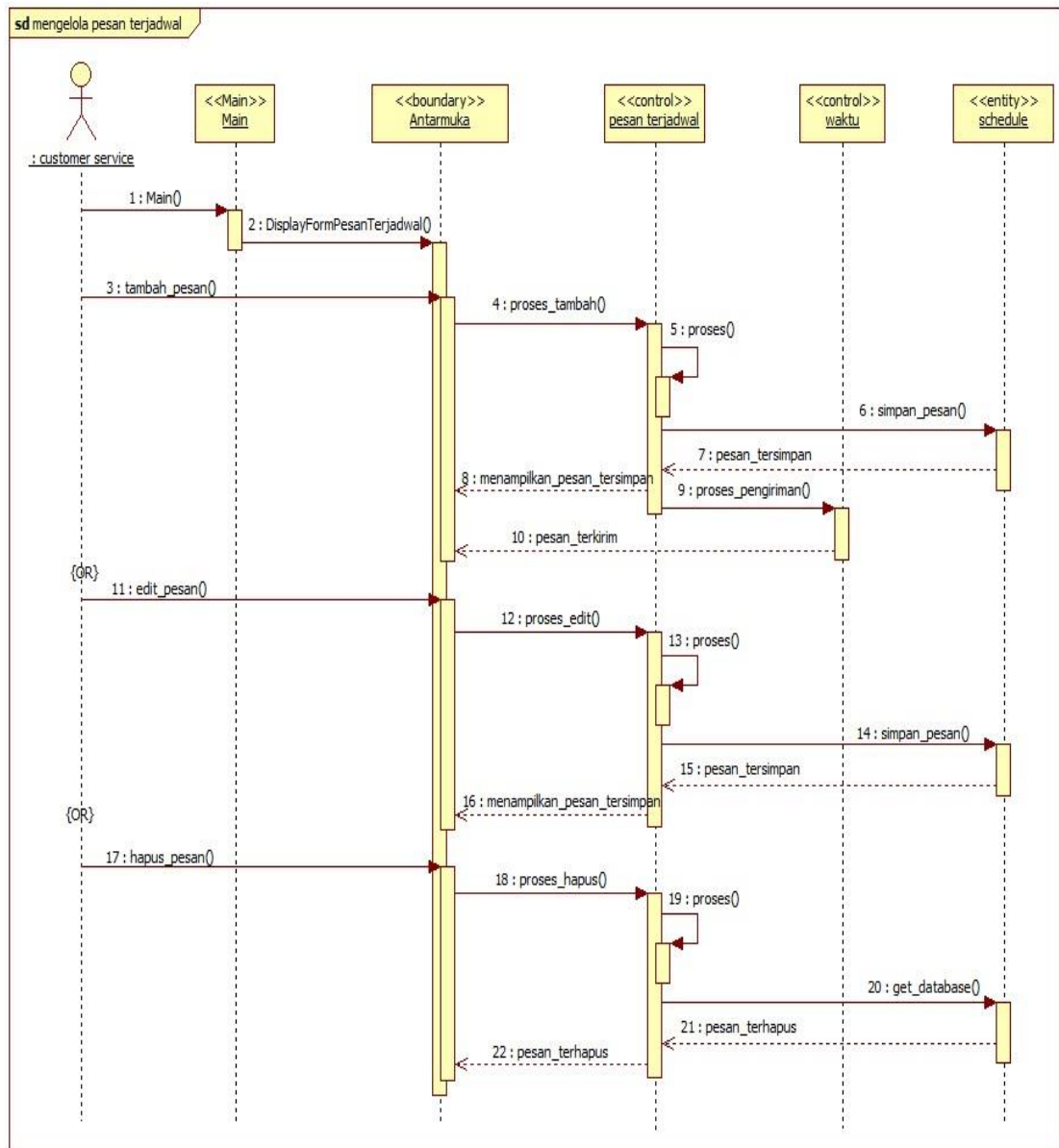
Pada gambar 3.30. menunjukkan *sequence diagram* pengiriman pesan tersiar, dimana proses *customer service* dalam mengirim sebuah pesan tersiar.



GAMBAR: 3.30. *Sequence Diagram* pengiriman pesan tersiar

4. *Sequence Diagram* mengelola pesan terjadwal

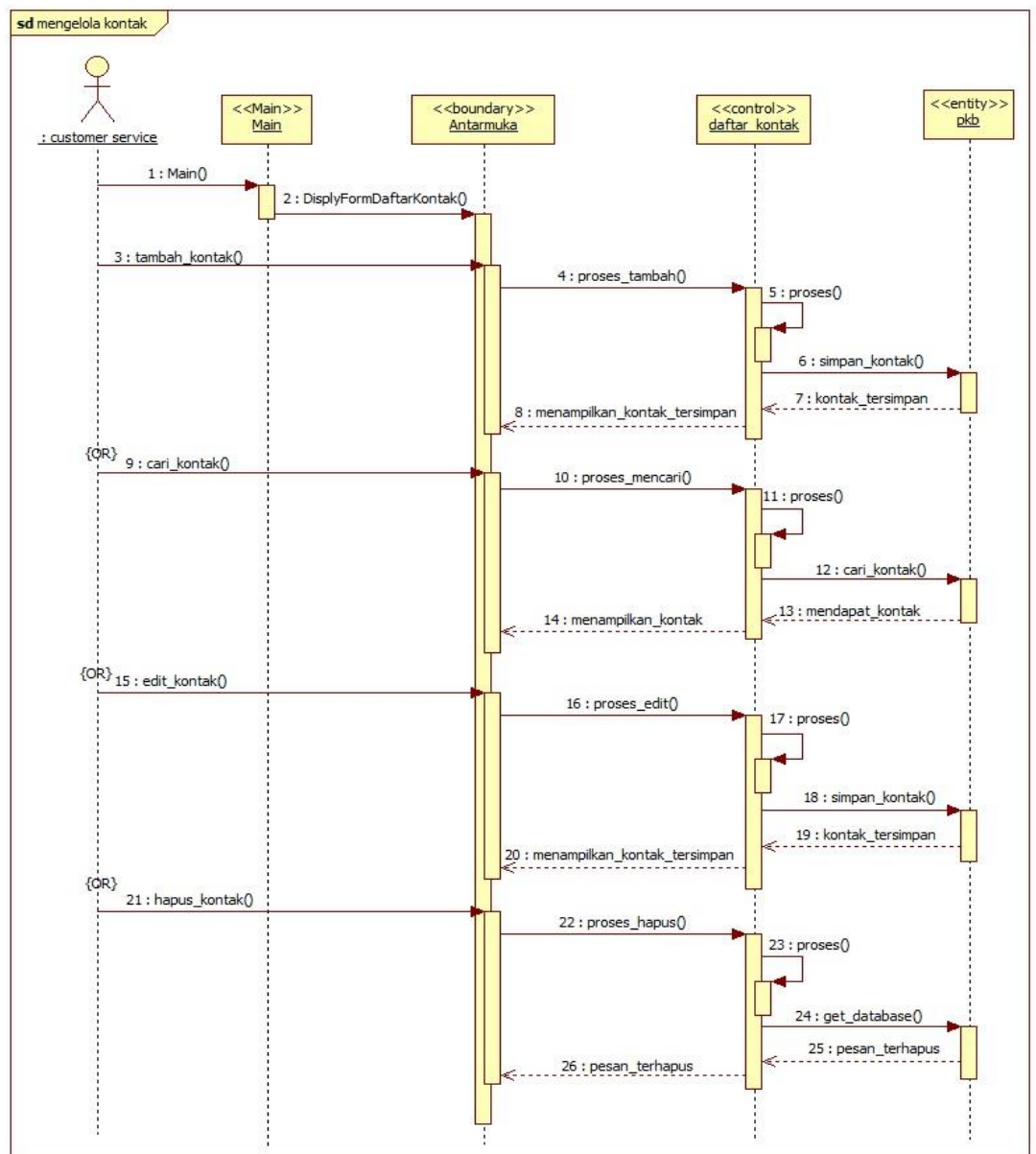
Pada gambar 3.31. menunjukkan *sequence diagram* mengelola pesan terjadwal, dalam *menu* mengelola pesan terjadwal terdapat beberapa proses yang di lakukan diantaranya proses tambah, proses edit dan proses hapus.



GAMBAR: 3.31. *Sequence Diagram* mengelola pesan terjadwal

5. Sequence Diagram mengelola kontak

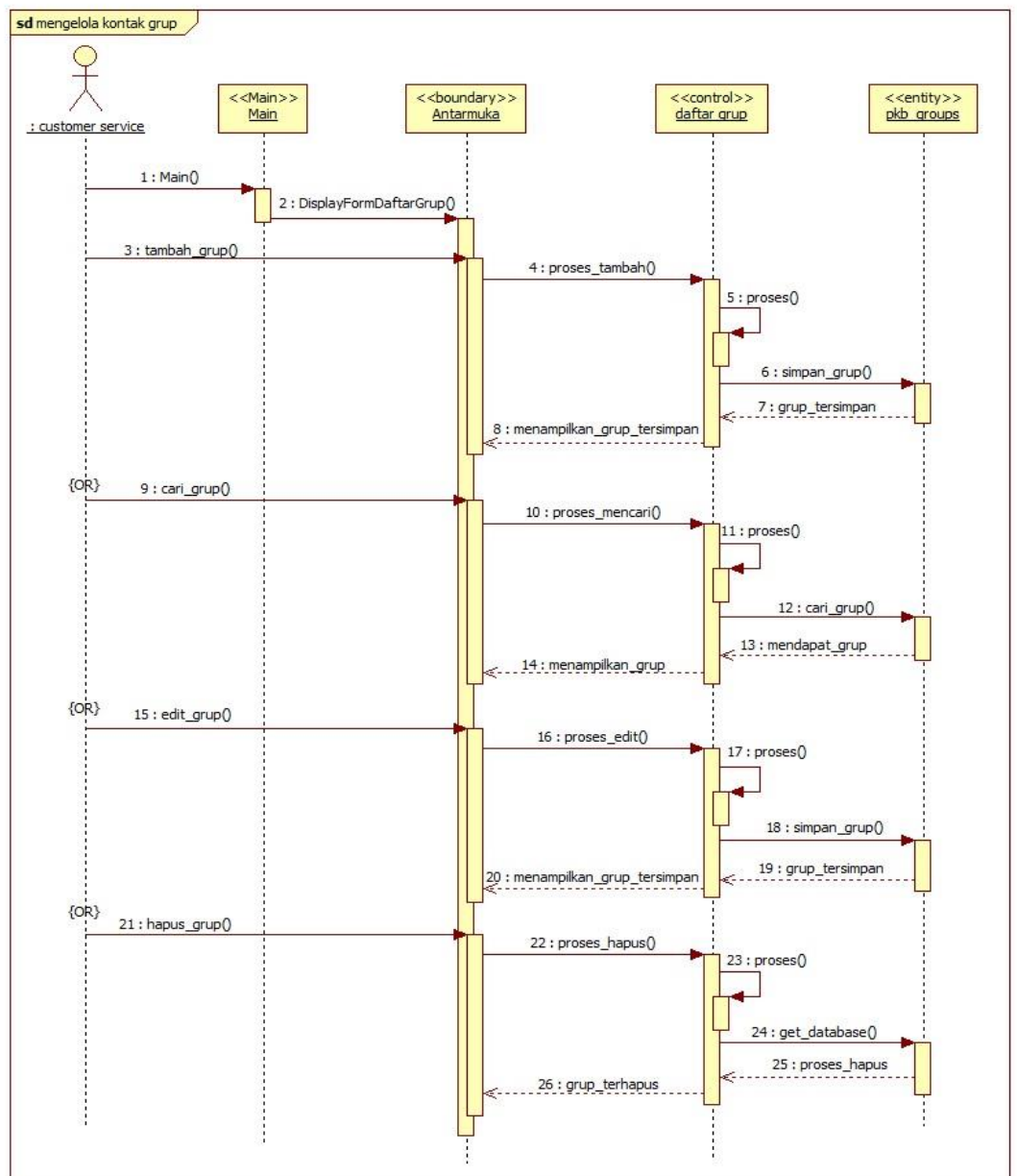
Pada gambar 3.32. menunjukkan *sequence diagram* mengelola kontak, dalam *menu* mengelola kontak terdapat beberapa proses yang dilakukan diantaranya proses tambah, proses cari, proses edit dan proses hapus.



GAMBAR: 3.32. *Sequence Diagram* mengelola kontak

6. Sequence Diagram mengelola kontak grup

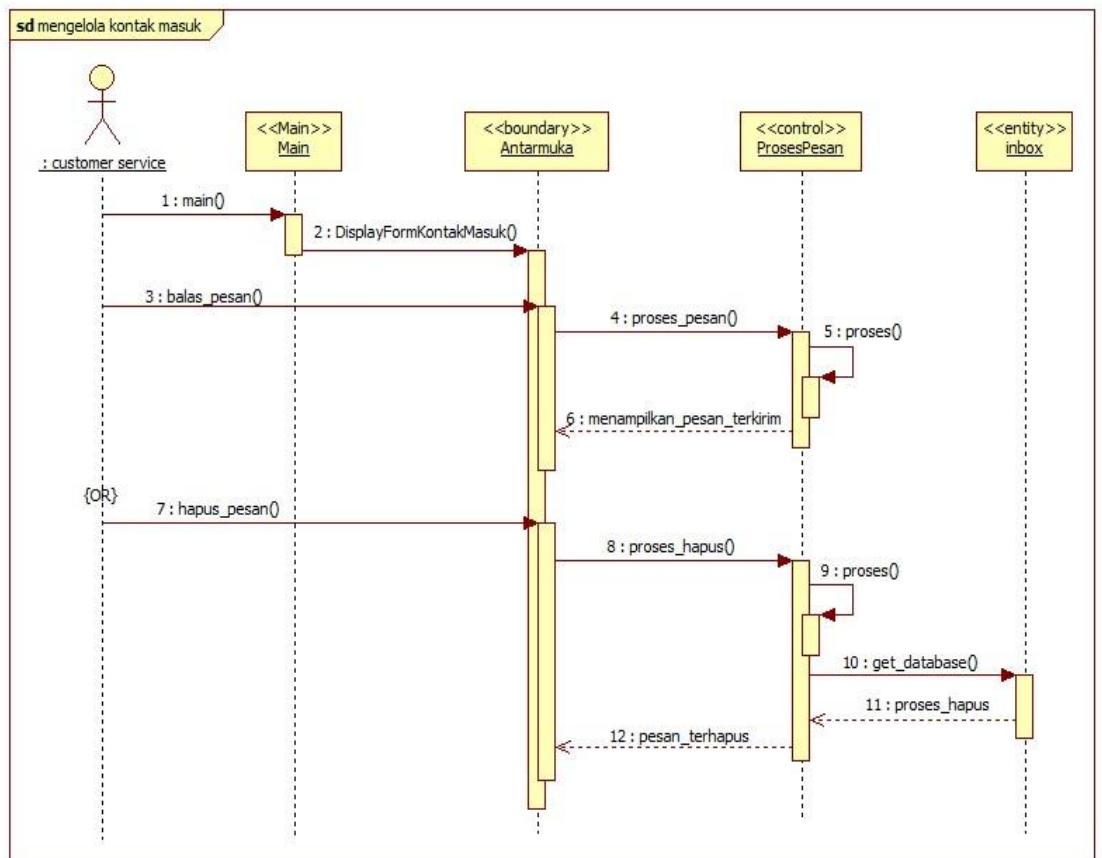
Pada gambar 3.33. menunjukkan *sequence diagram* mengelola kontak grup, dalam *menu* mengelola kontak grup terdapat beberapa proses yang di lakukan diantaranya proses tambah, proses cari, proses edit dan proses hapus.



GAMBAR: 3.33. Sequence Diagram mengelola kontak grup

7. Sequence Diagram mengelola kontak masuk

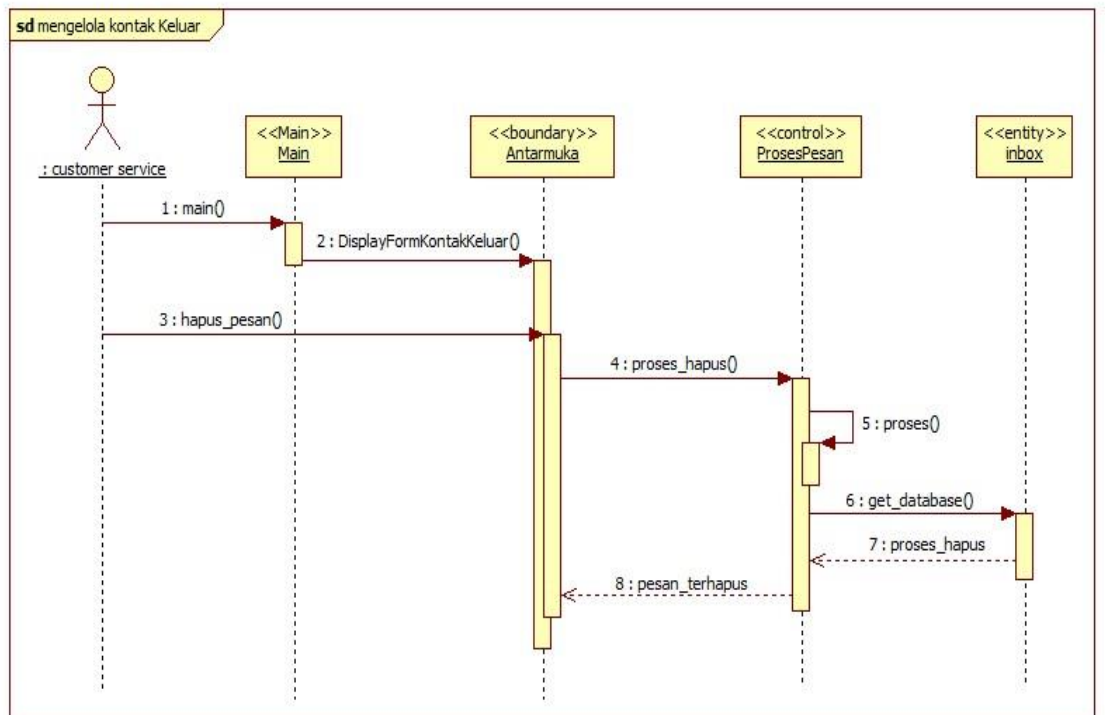
Pada gambar 3.34. menunjukkan *sequence diagram* mengelola kontak masuk, dalam *menu* mengelola kontak masuk terdapat beberapa proses yang di lakukan diantaranya proses balas dan proses hapus.



GAMBAR: 3.34. *Sequence Diagram* mengelola kontak masuk

8. *Sequence Diagram* mengelola kontak keluar

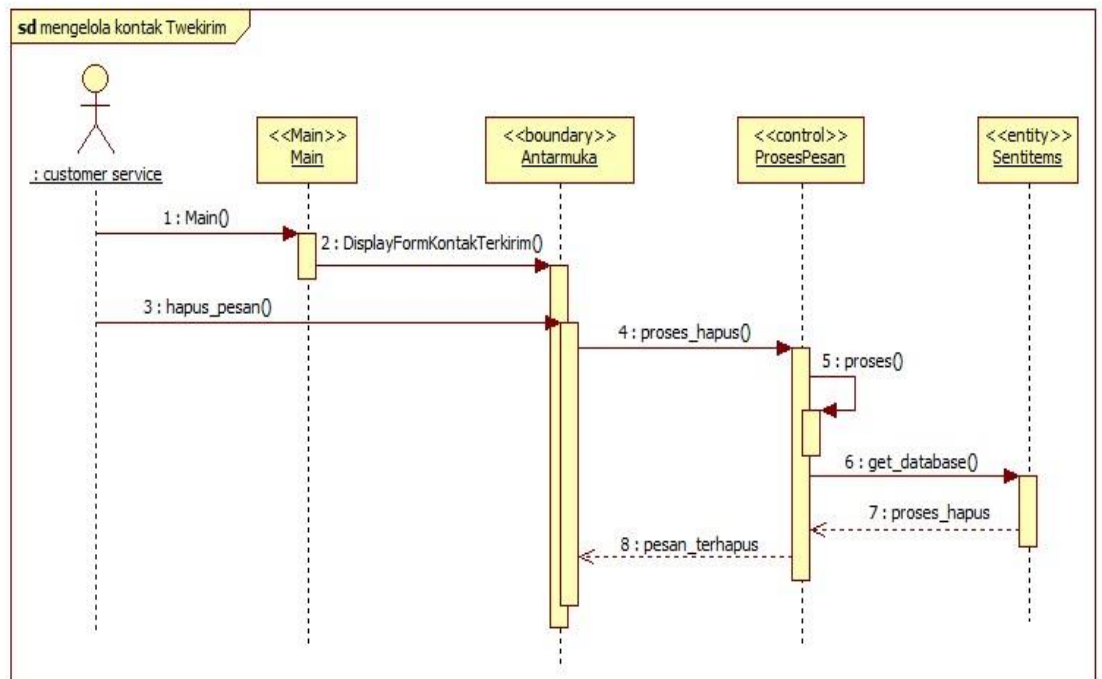
Pada gambar 3.35. menunjukkan *sequence diagram* mengelola kontak keluar, dalam *menu* mengelola kontak keluar terdapat proses yang di lakukan diantaranya proses hapus.



GAMBAR: 3.35. *Sequence Diagram* mengelola kontak keluar

9. *Sequence Diagram* mengelola kontak terkirim

Pada gambar 3.36. menunjukkan *sequence diagram* mengelola kontak terkirim, dalam *menu* mengelola kontak terkirim terdapat proses yang di lakukan diantaranya proses hapus.

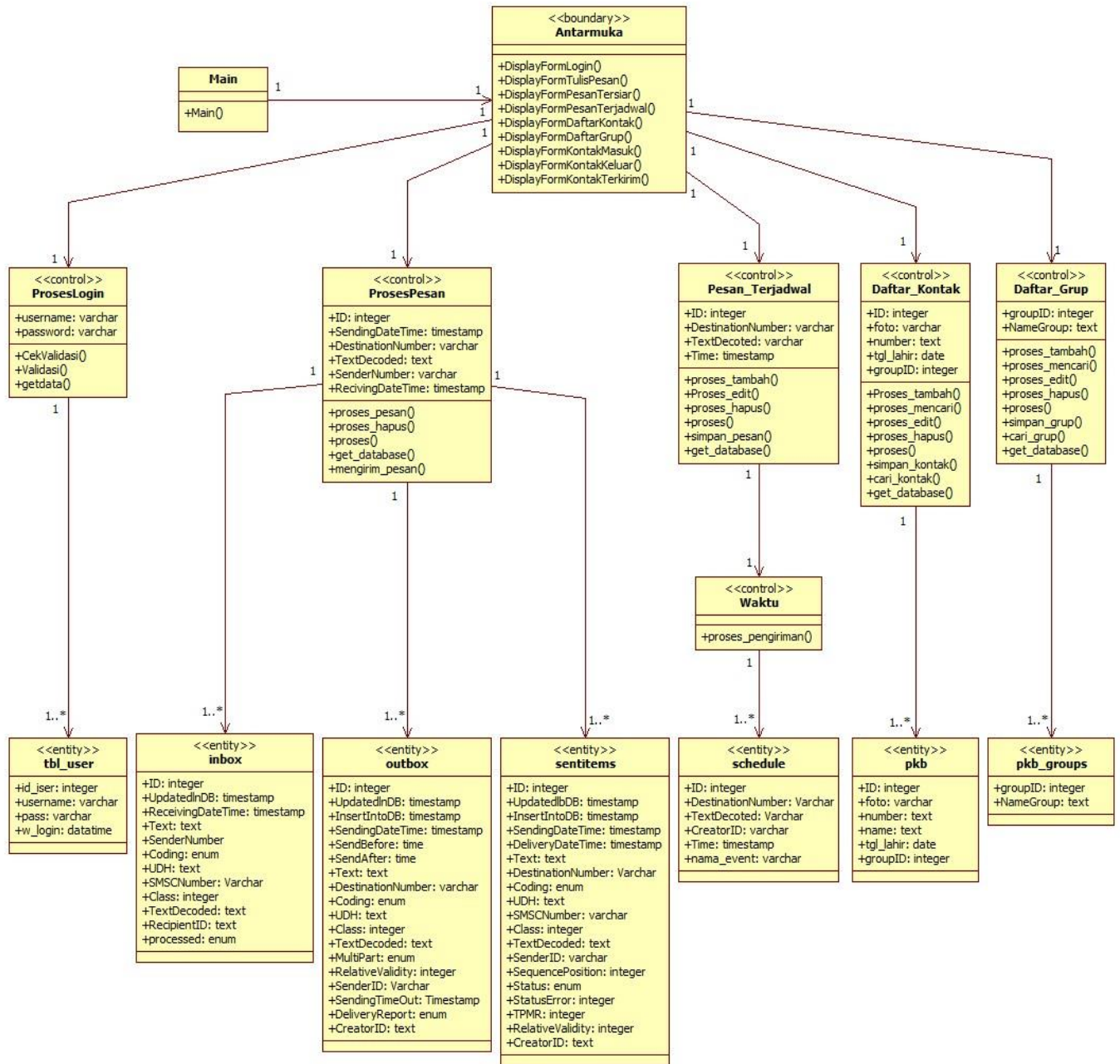


GAMBAR: 3.36. *Sequence Diagram* mengelola kontak terkirim

3.2.1.4. *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

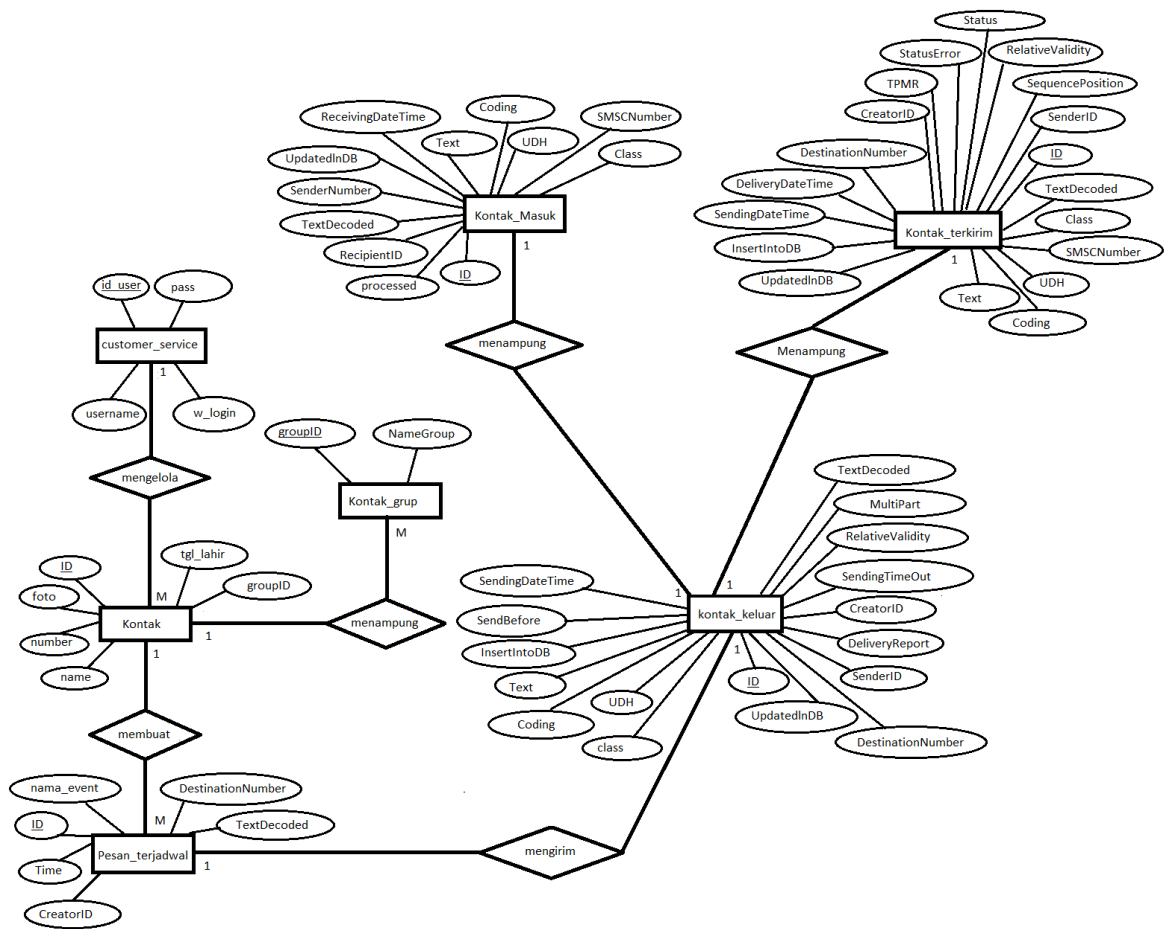
Kelas memiliki apa yang disebut atribut dan metode atau operasi.



GAMBAR: 3.37. Class diagram

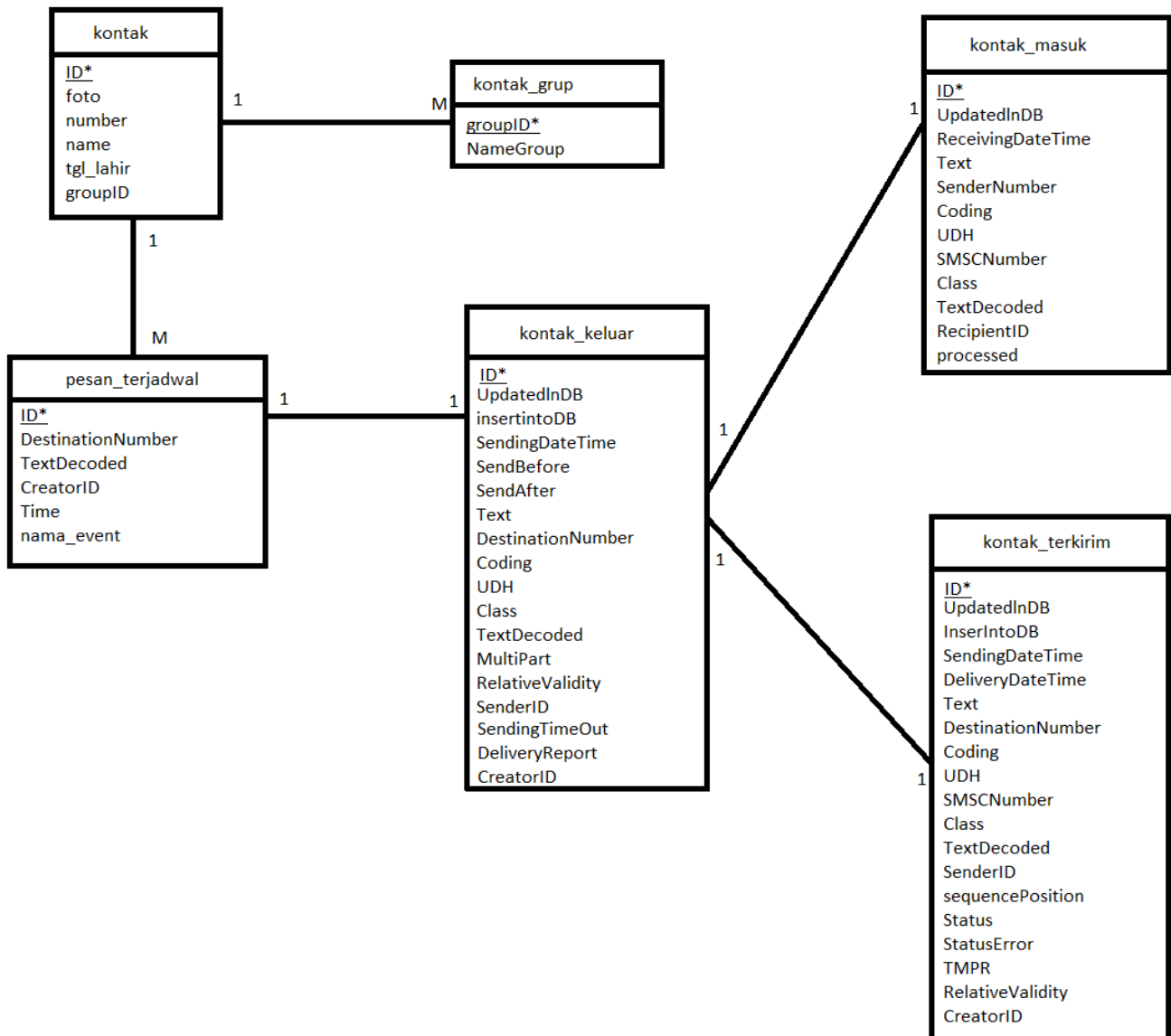
3.2.1.5. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah salah satu model yang digunakan untuk mendesain database dengan tujuan menggambarkan data yang berelasi pada sebuah database. Pada Gambar 3.38. Ini adalah Entity Relationship Diagram (ERD).



GAMBAR: 3.38. Entity Relationship Diagram

Gambar 3.39. menggambarkan relasi antar tabel yang ada dalam aplikasi ini. Seperti berikut ini:



GAMBAR: 3.39. Relasi antar tabel

Terdapat tujuh tabel yang terdapat pada perancang aplikasi untuk *customer service* berbasis *web* menggunakan *SMS Gateway* diantaranya adalah Tabel *customer service*, tabel kontak, tabel kontak grup, tabel pesan terjadwal, tabel kontak keluar, tabel kontak masuk dan tabel kontak terkirim.

TABEL: 3.28. Tabel *customer service*

Nama	Tipe Data	Ukuran	Key
Id_user	Integer	10	<i>Primary-Key</i>
Username	Varchar	100	-
Pass	Varchar	100	-
W_login	datetime	-	-

TABEL: 3.29. Tabel kontak

Nama	Tipe Data	Ukuran	Key
ID	integer	11	<i>Primary-Key</i>
Foto	varchar	250	-
Number	Text	-	-
Name	Text	-	-
Tgl_lahir	date	-	-
groupID	Integer	11	-

TABEL: 3.30. Tabel kontak grup

Nama	Tipe Data	Ukuran	Key
NameGroup	Text	-	<i>Primary-Key</i>
GroupID	Integer	11	-

TABEL: 3.31. Tabel pesan terjadwal

Nama	Tipe Data	Ukuran	Key
ID	Integer	3	<i>Primary-Key</i>
DestinationNumber	Varchar	160	-
TextDecoded	Varchar	50	-
CreatorID	Varchar	5	-
Time	Timestamp	-	-
Nama_event	Varchar	25	-

TABEL: 3.32. Tabel kontak masuk

Nama	Tipe Data	Ukuran	Key
ID	Integer	10	<i>Primary-Key</i>
UpdatedInDB	timestamp	-	-
ReceivingDateTime	Timestamp	-	-
Text	Text	-	-
SenderNumber	Varchar	20	-
Coding	Enum	-	-
UDH	Text	-	-
SMSCNumber	Varchar	20	-
Class	Integer	10	-
TextDecoded	Text	-	-
RecipientID	text	-	-
Processed	enum	-	-

TABEL: 3.33. Tabel kontak keluar

Nama	Tipe Data	Ukuran	Key
ID	Integer	10	<i>Primary-Key</i>
UpdatedInDB	timestamp	-	-
ReceivingDateTime	Timestamp	-	-
SendingDateTime	Timestamp	-	-
SendBefore	time	-	-
SendAfter	time	-	-
Text	Text	-	-
DestinationNumber	Varchar	20	-
Coding	Enum	-	-
UDH	Text	-	-
Class	Integer	11	-
TextDecoded	Text	-	-

MultiPart	enum	-	-
RelativeValidity	integer	11	-
SenderID	Varchar	255	-
SendingTimeOut	timestamp	-	-
DeliveryReport	enum	-	-
CreatorID	text	-	-

TABEL: 3.34. Tabel kontak terkirim

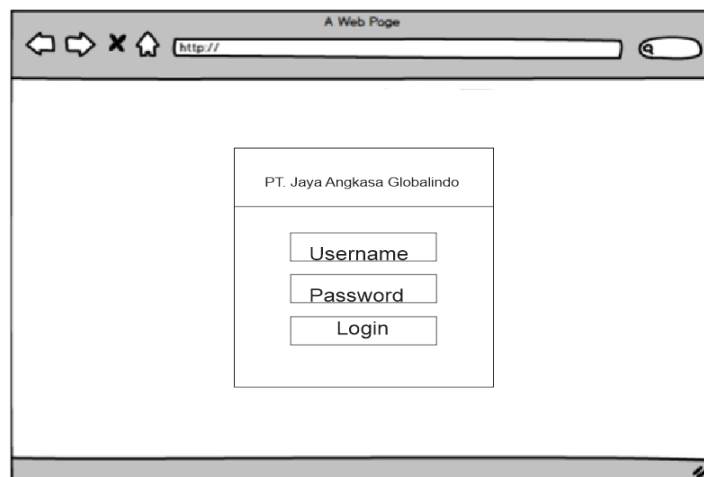
Nama	Tipe Data	Ukuran	Key
ID	Integer	10	<i>Primary-Key</i>
UpdatedInDB	timestamp	-	-
insertIntoDB	timestamp	-	-
SendingDateTime	Timestamp	-	-
DeliveryDateTime	Timestamp	-	-
Text	Text	-	-
DestinationNumber	Varchar	20	-
Coding	Enum	-	-
UDH	Text	-	-
SMSCNumber	Varchar	20	-
Class	Integer	11	-
TextDecoded	Text	-	-
SenderID	Varchar	255	-
SequencePosition	integer	11	-
Status	enum	-	-
StatusError	integer	11	-
TPMR	integer	11	-
RelativeValidity	integer	11	-
CreatorID	text	-	-

3.2.1.6. Design Interface

Design Interface merupakan rancangan antarmuka yang akan digunakan sebagai perantara *customer service* dengan perangkat lunak yang dikembangkan. *Design Interface* dari perancang aplikasi untuk *customer service* berbasis *web* menggunakan *SMS Gateway* sebagai berikut:

1. *Design interface form login*

Gambar 3.40 *Form login* merupakan tampilan halaman pertama membuka *web* untuk menginput *username* dan *password* agar dapat masuk ke halaman *dashboard*.

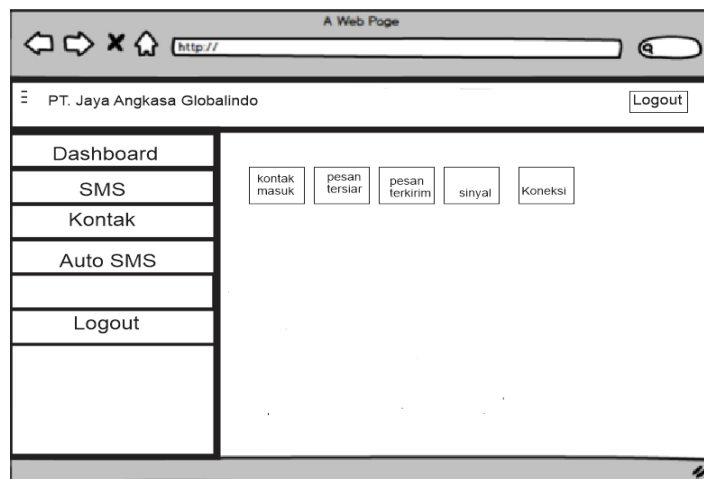


The image shows a web browser window titled "A Web Page". The address bar contains "http://". The main content area displays a login form for "PT. Jaya Angkasa Globalindo". The form consists of three input fields: "Username", "Password", and a "Login" button.

GAMBAR: 3.40. *Design interface Form Login*

2. *Design interface dashboard*

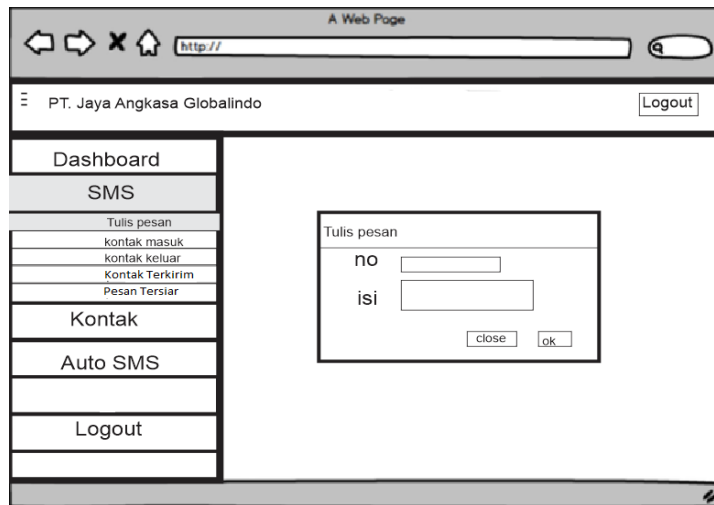
Gambar 3.41 *dashboard* merupakan tampilan halaman utama ketika *customer service* sudah melakukan *login*, *customer service* dapat melakukan aktivitas seperti mengirim pesan, mengirim pesan tersiar, mengelola pesan terjadwal, mengelola kontak, mengelola kontak grup, mengelola kontak masuk, mengelola kontak keluar dan mengelola kontak terkirim.



GAMBAR: 3.41. *Design interface Dashboard*

3. *Design interface tulis pesan*

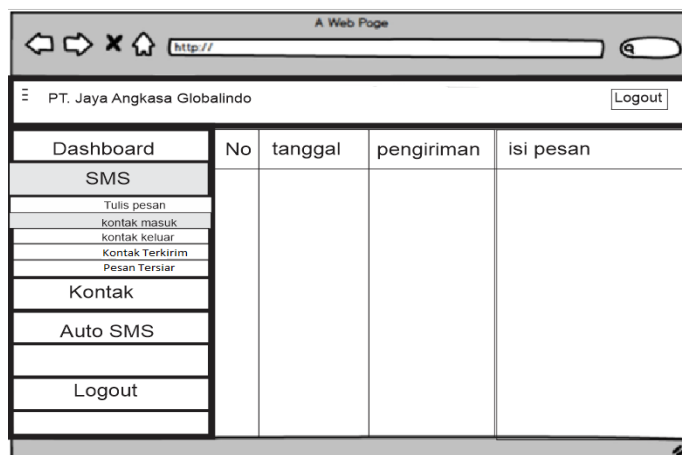
Gambar 3.42 tulis pesan merupakan tampilan untuk mengirim sebuah pesan kepada *customer* dengan menginput pesan dan kontak yang kemudian pesan akan dikirimkan kepada *customer*.



GAMBAR: 3.42. *Design interface* Tulis pesan

4. *Design interface* kontak masuk

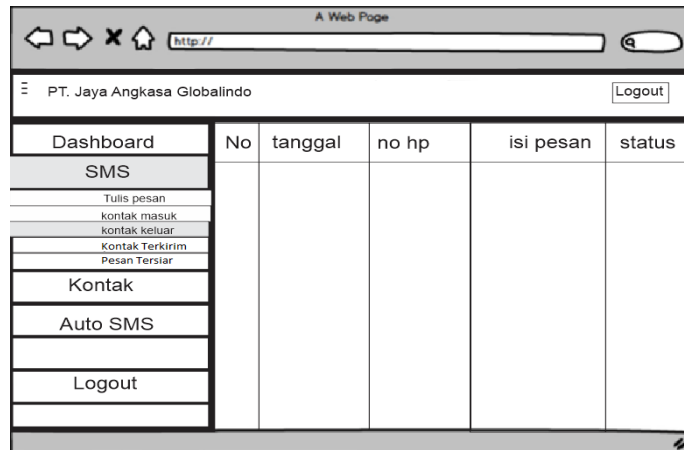
Gambar 3.43 kontak masuk merupakan tampilan untuk menerima sebuah pesan dari *customer*, pesan akan tersimpan di kontak masuk ketika *customer* mengirim sebuah pesan.



GAMBAR: 3.43. *Design interface* Kontak masuk

5. *Design interface* kontak keluar

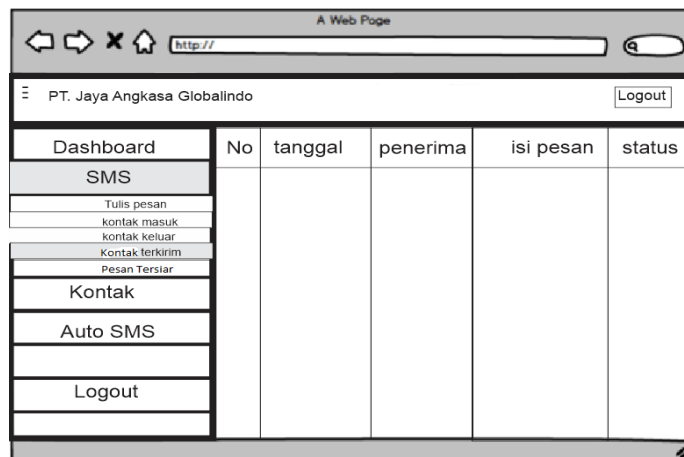
Gambar 3.44 kontak keluar merupakan tampilan untuk memproses sebuah pesan yang belum terkirim, pesan akan di simpan sementara di kontak keluar sebelum pesan itu terkirim.



GAMBAR: 3.44. *Design interface* Kontak keluar

6. *Design interface* kontak terkirim

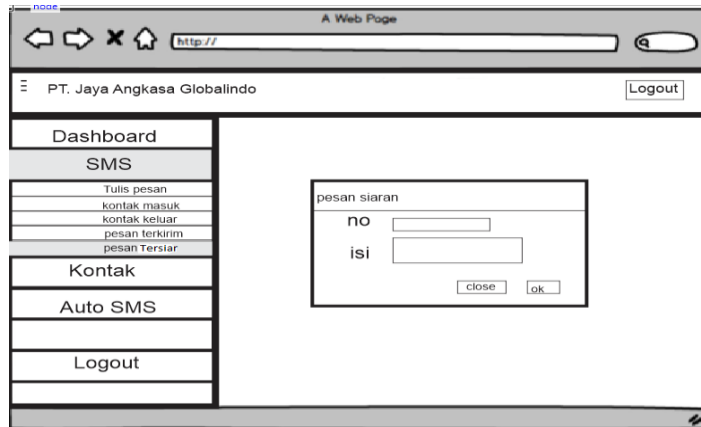
Gambar 3.45 kontak terkirim merupakan tampilan untuk memproses sebuah pesam yang sudah berhasil terkirim.



GAMBAR: 3.45. *Design interface* Kontak Terkirim

7. Design interface pesan tersiar

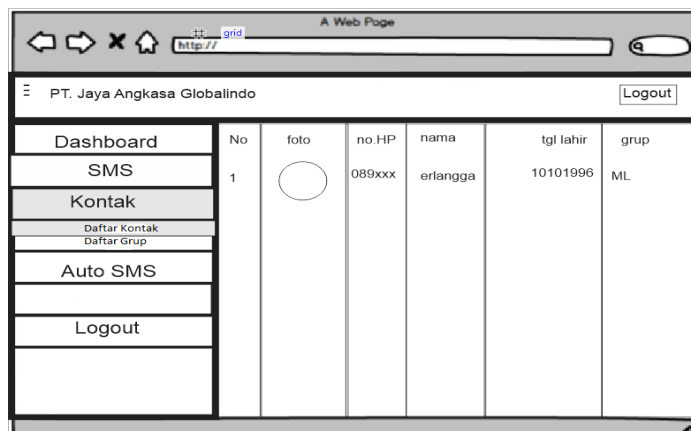
Gambar 3.46 pesan tersiar merupakan tampilan untuk mengirim sebuah pesan tersiar atau *send all* kepada *customer* dengan menginput pesan dan kontak yang kemudian pesan akan terkirim keseluruh *customer*.



GAMBAR: 3.46. Design interface Pesan Tersiar

8. Design interface daftar kontak

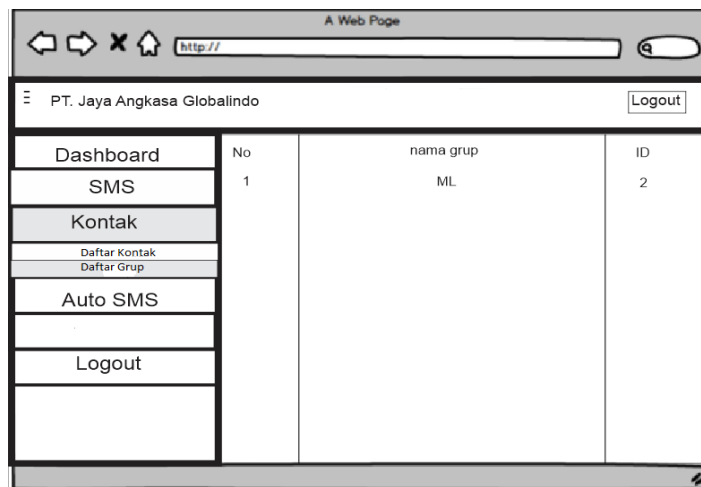
Gambar 3.47 daftar kontak merupakan tampilan untuk mengelola sebuah kontak, *customer service* dapat melakukan aktivitas seperti menambah kontak, menghapus kontak dan mengedit kontak.



GAMBAR: 3.47. Design interface daftar kontak

9. *Design interface* daftar grup

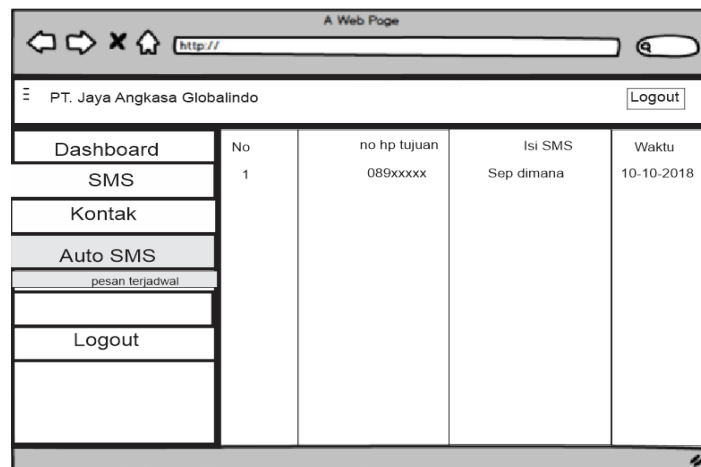
Gambar 3.48 daftar kontak grup merupakan tampilan untuk mengelola sebuah kontak grup, *customer service* dapat melakukan aktivitas seperti menambah kontak grup, menghapus kontak grup dan mengedit kontak grup.



GAMBAR: 3.48. *Design interface* Daftar Grup

10. *Design interface* pesan terjadwal

Gambar 3.49 pesan terjadwal merupakan tampilan untuk mengelola pesan secara terjadwal dengan menginput pesan dan kontak yang kemudian pesan akan terkirim keseluruh *customer*. *customer service* dapat melakukan aktivitas seperti menambah kontak, menghapus kontak dan mengedit kontak.



GAMBAR: 3.49. *Design interface* Pesan Terjadwal

3.3. Pengodean

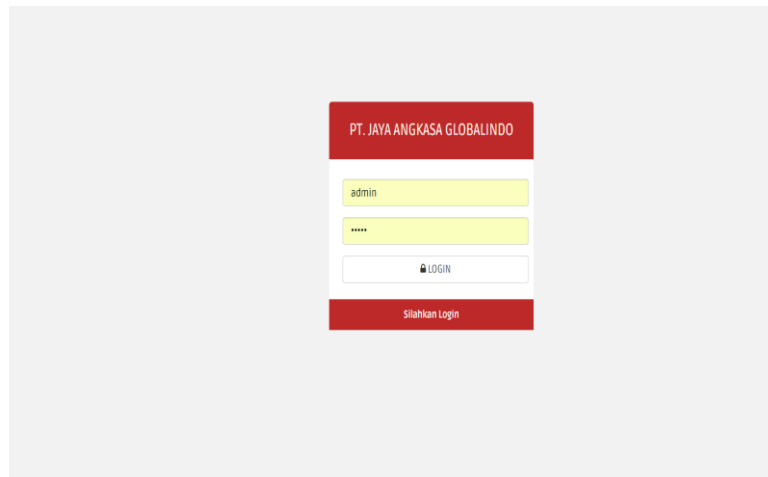
Setelah sistem dianalisis dan didesain, maka selanjutnya menuju tahap pengodean. Pengodean merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Pengodean dari perancang aplikasi untuk *customer service* berbasis *web* menggunakan *SMS Gateway*.

3.3.1. Implementasi

Berikut adalah tampilan perancang aplikasi untuk *customer service* berbasis *web* menggunakan *SMS Gateway* yang dibuat pada tahap sebelumnya.

1. Tampilan *form login*

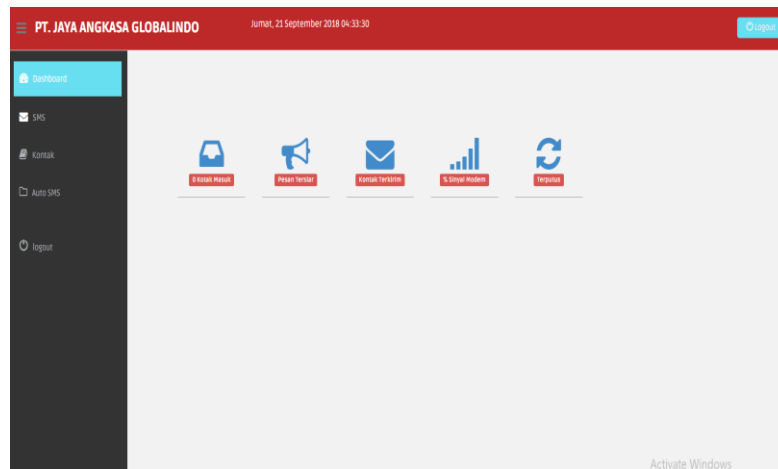
Halaman login merupakan halaman awal dari customer service sebelum masuk ke halaman utama customer service. Adapun tampilan halaman login sebagai berikut :



GAMBAR: 3.50. Tampilan implementasi *form login*

2. *Dashboard*

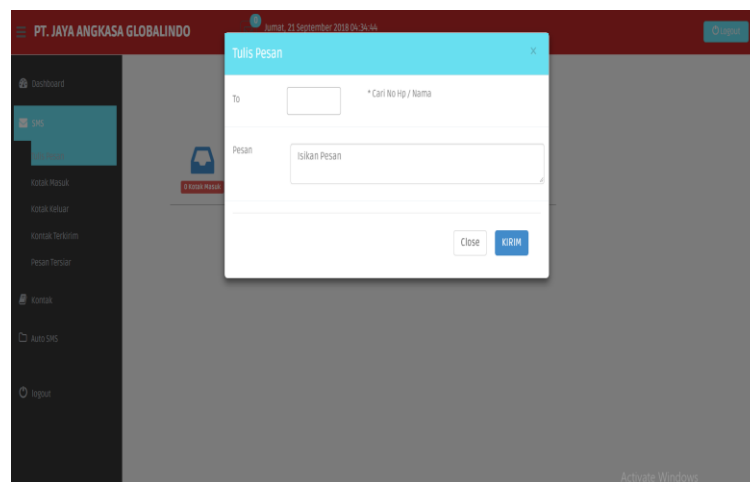
Halaman *dashboard* merupakan tampilan halaman utama ketika *customer service* sudah melakukan *login*, *customer service* dapat melakukan aktivitas seperti mengirim pesan, mengirim pesan tersiar, mengelola pesan terjadwal, mengelola kontak, mengelola kontak grup, mengelola kontak masuk, mengelola kontak keluar dan mengelola kontak terkirim. Adapun tampilan sebagai berikut :



GAMBAR: 3.51. Tampilan implementasi *Dashboard*

3. Tampilan tulis pesan

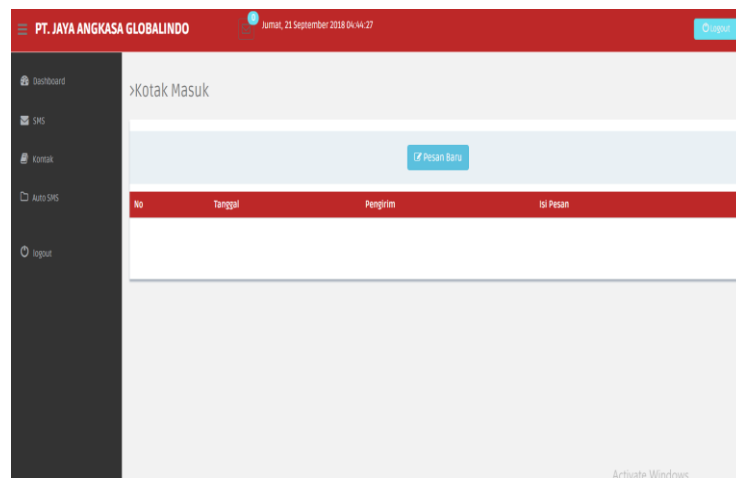
Halaman tulis pesan merupakan tampilan untuk mengirim sebuah pesan kepada *customer* dengan menginput pesan dan kontak yang kemudian pesan akan dikirimkan kepada *customer*. Adapun tampilan sebagai berikut :



GAMBAR: 3.52. Tampilan implementasi tulis pesan

4. Tampilan Kontak Masuk

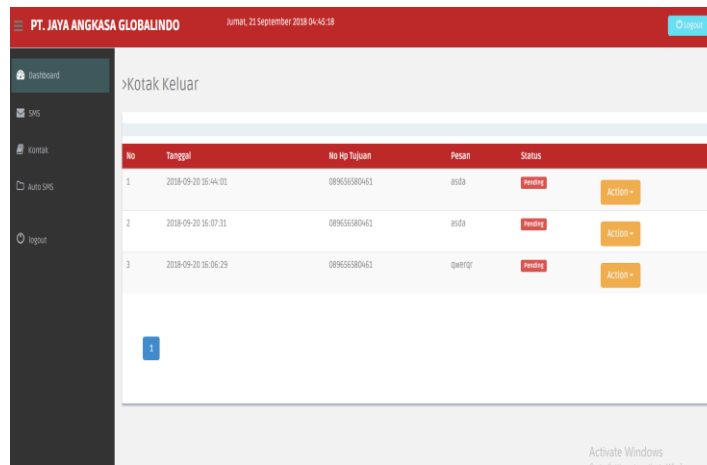
Halaman kontak masuk merupakan tampilan untuk menerima sebuah pesan dari *customer*, pesan akan tersimpan di kontak masuk ketika *customer* mengirim sebuah pesan. Adapun tampilan sebagai berikut :



GAMBAR: 3.53. Tampilan implementasi Kontak Masuk

5. Tampilan Kontak Keluar

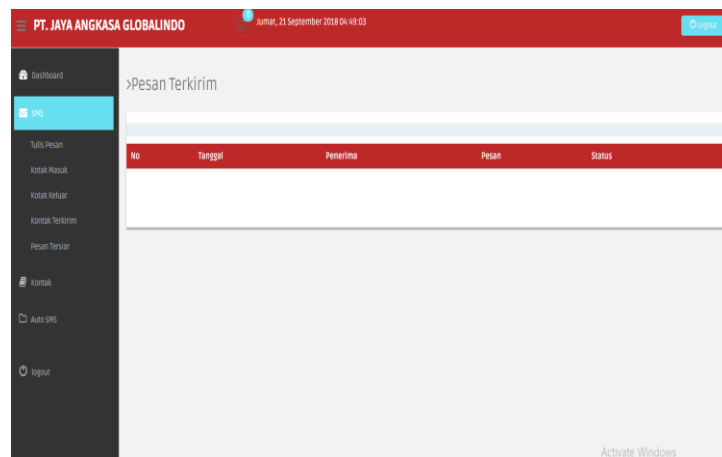
Halaman kontak keluar merupakan tampilan untuk memproses sebuah pesan yang belum terkirim, pesan akan di simpan sementara di kontak keluar sebelum pesan itu terkirim. Adapun tampilan sebagai berikut :



GAMBAR: 3.54. Tampilan implementasi Kontak Keluar

6. Tampilan kontak Terkirim

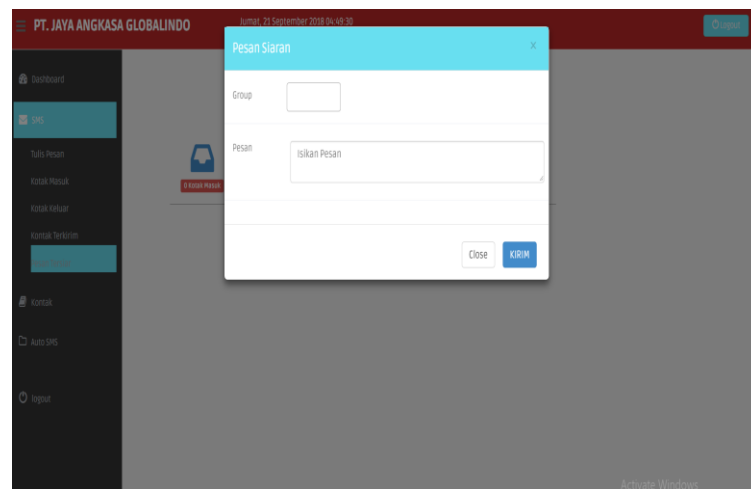
Halaman kontak terkirim merupakan tampilan untuk memproses sebuah pesan yang sudah berhasil terkirim. Adapun tampilan sebagai berikut :



GAMBAR: 3.55. Tampilan implementasi kontak Terkirim

7. Tampilan Pesan Tersiar

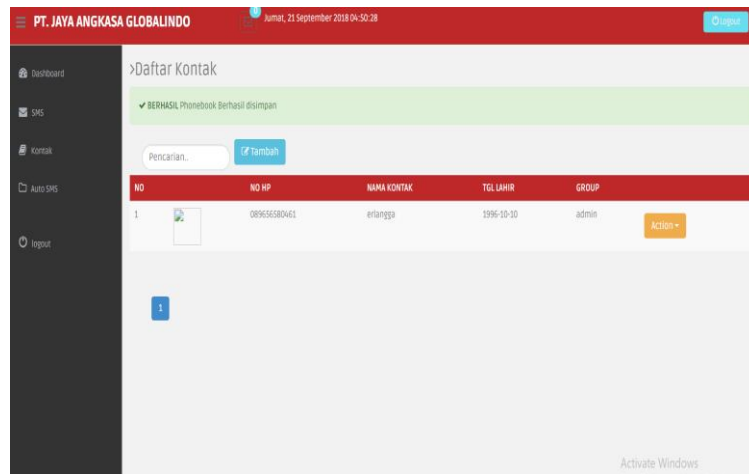
Halaman pesan tersiar merupakan tampilan untuk mengirim sebuah pesan tersiar atau *send all* kepada *customer* dengan menginput pesan dan kontak yang kemudian pesan akan terkirim keseluruhan *customer*. Adapun tampilan sebagai berikut :



GAMBAR: 3.56. Tampilan implementasi Pesan Tersiar

8. Tampilan daftar kontak

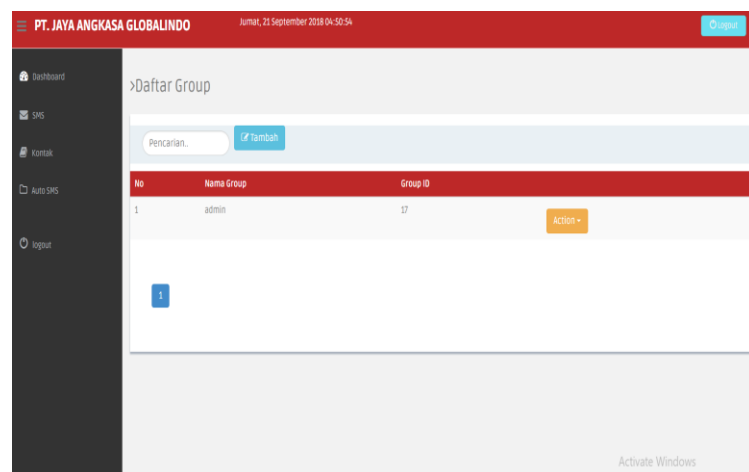
Halaman daftar kontak merupakan tampilan untuk mengelola sebuah kontak, *customer service* dapat melakukan aktivitas seperti menambah kontak, menghapus kontak dan mengedit kontak. Adapun tampilan sebagai berikut :



GAMBAR: 3.57. Tampilan implementasi daftar kontak

9. Tampilan Mengelola daftar grup

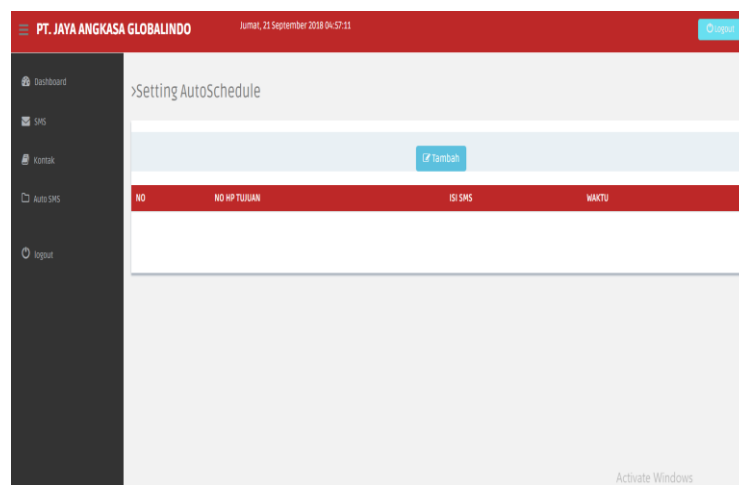
Halaman daftar kontak grup merupakan tampilan untuk mengelola sebuah kontak grup, *customer service* dapat melakukan aktivitas seperti menambah kontak grup, menghapus kontak grup dan mengedit kontak grup. Adapun tampilan sebagai berikut :



GAMBAR: 3.58. Tampilan implementasi daftar grup

10. Tampilan Pesan Terjadwal

Halaman pesan terjadwal merupakan tampilan untuk mengelola pesan secara terjadwal dengan menginput pesan dan kontak yang kemudian pesan akan terkirim keseluruh *customer*. *customer service* dapat melakukan aktivitas seperti menambah kontak, menghapus kontak dan mengedit kontak. Adapun tampilan sebagai berikut :



GAMBAR: 3.59. Tampilan implementasi Pesan Terjadwal

11. Tampilan mengirim pesan ke *customer*

Tampilan ketika *customer service* mengirim sebuah pesan kepada *customer*. Adapun tampilan sebagai berikut :



GAMBAR: 3.60. Tampilan implementasi mengirim pesan ke *customer*

3.4. Pengujian

Pengujian yang akan digunakan dalam menguji sistem adalah dengan metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

3.4.1. Rencana Pengujian

Rencana Pengujian dapat dilihat pada tabel 3.35.

TABEL: 3.35. Rencana Pengujian aplikasi

No	Item Uji	Detail Pengujian
1	<i>Login</i>	Verifikasi Data <i>Login</i>
2	pengiriman pesan	Mengirim pesan
3	pengiriman pesan tersiar	Mengirim pesan tersiar atau <i>sendall</i>
4	Mangelola pesan terjadwal	Tambah pesan terjadwal
		Edit pesan terjadwal
		Hapus pesan terjadwal
5	Mangelola kontak	tambah kontak
		Edit kontak
		Cari kontak
		Hapus kontak
6	Mangelola kontak grup	Tambah grup
		Edit grup
		cari grup
		Hapus grup
7	Mangelola kontak Masuk	Balas kontak masuk
		Hapus kontak masuk
8	Mangelola kontak keluar	Hapus kontak keluar
9	Mengelola kontak terkirim	Hapus kontak terkirim

3.4.2. Kasus dan Hasil Pengujian

Berdasarkan rencan Pengujian yang telah disusun, maka dapat dilakukan pengujian seperti pada tabel 3.36.

TABEL: 3.36. Pengujian perancangan aplikasi

No.	Item uji	Data masukan	Hasul yang diharapkan	Hasil pengujian	kesimpulan
1.	Login	Username: Admin Password: Admin	Sistem Menampilkan halaman <i>dashboard</i> sesuai dengan hak akses	Sesuai harapan	Valid
		Username: - Password: -	Sistem menampilkan informasi “gagal login” mohon isi <i>username</i> dan <i>password</i>	Sesuai harapan	Valid
		Username: Admin Password: -	Sistem Menampilkan informasi “gagal login” mohon isi <i>password</i>	Sesuai harapan	Valid
		Username: - Password: Admin	Sistem Menampilkan informasi “gagal login” mohon isi <i>username</i>	Sesuai harapan	Valid
		Username: amin Password: Admin	Sistem Menampilkan informasi “gagal login” dan mereset ulang	Sesuai harapan	Valid
2.	pengiriman pesan	menginput kontak dan <i>input</i> pesan kemudian klik kirim	Sistem menampilkan pesan berhasil di kirim	Sesuai harapan	Valid
		Tidak menginput apapun kemudian klik kirim	Sistem menampilkan “harap isi”	Sesuai harapan	Valid
		Tidak menginput pesan kemudian kirim	Sistem menampilkan “harap isi”	Sesuai harapan	Valid
		Tidak menginput kontak kemudian klik kirim	Sistem menampilkan “harap isi”	Sesuai harapan	Valid

3.	Pengiriman pesan tersiar	menginput kontak grup dan <i>input</i> pesan kemudian klik kirim	Sistem menampilkan pesan berhasil di kirim	Sesuai harapan	Valid
		Tidak menginput apapun kemudian klik kirim	Sistem menampilkan “harap isi”	Sesuai harapan	Valid
		Tidak menginput pesan kemudian klik kirim	Sistem menampilkan “harap isi”	Sesuai harapan	Valid
		Tidak menginput kontak grup kemudian klik kirim	Sistem menampilkan “harap isi”	Sesuai harapan	Valid
4.	Tambah pesan terjadwal	Menginput semua <i>field</i> data dan kemudian klik simpan	Tersimpan pesan terjadwal dan akan terkirim otomatis sesuai waktu pengiriman pesan sudah di tentukan	Sesuai harapan	Valid
		Tidak menginput apapun kemudian klik simpan	Sistem menampilkan “pesan harus di isi”	Sesuai harapan	Valid
		Menginput semua <i>field</i> data kecuali tidak menginput pesan dan klik simpan	Sistem menampilkan “pesan harus di isi”	Sesuai harapan	Valid
		Menginput semua <i>field</i> data kecuali tidak menginput waktu kirim dan klik simpan	Sistem menampilkan “waktu harus di isi”	Sesuai harapan	Valid
		Menginput semua <i>field</i> data kecuali tidak menginput kontak atau grup dan klik simpan	Sistem menampilkan “masukan kontak”	Sesuai harapan	Valid

5.	Edit pesan terjadwal	Melakukan edit dengan data yang sesuai	Sistem menampilkan perubahan <i>update</i> terbaru	Sesuai harapan	Valid
6.	Hapus pesan terjadwal	Klik tombol hapus pada pesan terjadwal yang akan di hapus	Sistem akan menghapus pesan terjadwal	Sesuai harapan	Valid
7.	Tambah kontak	Menginput semua <i>field</i> data dan kemudian klik tambah	Sistem akan menampilkan " <i>phonebook</i> berhasil disimpan"	Sesuai harapan	Valid
		Tidak Menginput semua <i>field</i> lalu klik tambah	Sistem akan menampilkan "harap isi bidang ini"	Sesuai harapan	Valid
		Menginput semua <i>field</i> kecuali nama dan kemudian klik tambah	Sistem akan menampilkan "harap isi bidang ini"	Sesuai harapan	Valid
		Menginput semua <i>field</i> kecuali nomor dan kemudian klik tambah	Sistem akan menampilkan "harap isi bidang ini"	Sesuai harapan	Valid
		Menginput semua <i>field</i> kecuali tanggal lahir dan kemudian klik tambah	Sistem akan menampilkan "harap isi bidang ini"	Sesuai harapan	Valid
8.	Cari kontak	<i>Keyword</i> nama kontak yang sudah tersimpan	Sistem menampilkan kontak yang sudah dibuat	Sesuai harapan	Valid
		<i>Keyword</i> nama kontak yang di cari	Sistem menampilkan tidak menemukan	Sesuai harapan	Valid
9.	Edit kontak	Melakukan edit dengan data yang sesuai	Sistem akan menampilkan perubahan <i>update</i> terbaru	Sesuai harapan	Valid

10.	Hapus kontak	Klik tombol hapus pada kontak yang akan di hapus	Sistem akan menghapus kontak	Sesuai harapan	Valid
11.	Tambah grup	Menginput nama grup dan kemudian klik tambah	Sistem akan menampilkan grup yang di buat	Sesuai harapan	Valid
		Tidak Menginput nama grup klik tambah	Sistem akan menampilkan “harap isi bidang ini”	Sesuai harapan	Valid
12.	Cari grup	<i>Keyword</i> nama kontak grup yang sudah tersimpan	Sistem menampilkan kontak yang sudah dibuat	Sesuai harapan	Valid
		<i>Keyword</i> nama kontak grup yang di cari	Sistem menampilkan tidak menemukan	Sesuai harapan	Valid
13.	Edit grup	Melakukan edit dengan data yang sesuai	Sistem akan menampilkan perubahan <i>update</i> terbaru	Sesuai harapan	Valid
14.	Hapus grup	Klik tombol hapus pada kontak grup yang akan di hapus	Sistem akan menghapus kontak grup	Sesuai harapan	Valid
15.	Balas kontak masuk	Mengisi <i>field</i> data text dan klik kirim	Pesan berhasil di kirim	Sesuai harapan	Valid
		Tidak mengisi <i>field</i> data text klik kirim	Sistem menampilkan “harap isi bidang ini”	Sesuai harapan	Valid
16.	Hapus kontak masuk	Klik tombol hapus pada pesan yang akan dihapus	Menampilkan notif pesan berhasil di hapus	Sesuai harapan	Valid

17.	Hapus kontak keluar	Klik tombol hapus pada pesan yang akan dihapus	Menampilkan notif pesan berhasil di hapus	Sesuai harapan	Valid
18.	Hapus kontak Terkirim	Klik tombol hapus pada pesan yang akan dihapus	Menampilkan notif pesan berhasil di hapus	Sesuai harapan	Valid

BAB IV

KESIMPULAN DAN SARAN

4.1. Kesimpulan

Kesimpulan yang diperoleh dari penulisan membuat perancangan aplikasi untuk *customer service* berbasis *web* menggunakan *SMS gateway* (Studi kasus di PT.Jaya Angkasa Globalindo) adalah sebagai berikut:

1. Dengan menerapkan aplikasi ini di PT. Jaya Angkas Globalindo dapat meningkatkan hubungan yang lebih baik pihak perusahaan dengan *customer*.
2. Aplikasi yang dibuat ini untuk *customer service* yang dapat memudahkan dalam menyampaikan sebuah informasi kepada *customer* di PT. Jaya Angkasa Globalindo, dan *customer* dapat mengakses informasi yang dibutuhkan lewat telepon seluler dengan format SMS.
3. Untuk dapat melakukan pengiriman pesan secara cepat di perlukan komputerisasi agar tidak memakan waktu yang lama dalam pengiriman informasi sehingga tidak perlu mengirim sms satu-persatu ataupun manual.

4.2. Saran

Ada pun saran yang diperoleh dari penulisan perancangan aplikasi untuk *customer service* berbasis *web* menggunakan *SMS gateway* (Studi kasus di PT.Jaya Angkasa Globalindo) adalah sebagai berikut:

1. Untuk pengembangan *SMS gateway* dalam peningkatan hubungan baik penangkal petir dengan *customer* maka perlu fitur *SMS gateway* yang lebih lengkap dan teknologi yang lebih baik disertai dengan kemampuan personal dari karyawan PT. Jaya Angkasa Globalindo, sehingga dengan demikian dapat menghasilkan sistem SMS yang lebih maksimal.
2. agar aplikasi ini tidak hanya berbasis SMS tetapi juga dapat berjalan pada aplikasi tidak berbayar seperti *what's app* dan juga *line*.
3. Diharapkan dalam pengembangan pembuatan aplikasi *SMS Gateway*, dapat menggunakan *tools* lain selain *gammu* sebagai *gateway*-nya.

DAFTAR PUSTAKA

- Aradea, R A. Dewanto. (2007). *Aplikasi Sms Gateway Dengan Koreksi Kesalahan Menggunakan Fuzzy String Matching*. Yogyakarta.
- Ariana, S., & Shalahuddin, M. (2014). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Ariana, S., & Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Azhar Susanto. (2013). *Kumpulan Materi Sistem*. Yogyakarta.
- Elfi Nur, & Wangdra Yvonne. (2016). *Rekayasa Perangkat Lunak*, Jakarta: Ilmu Teknologi.
- Enterprise Jubilee. (2016). *Pengenalan HTML dan CSS*. Jakarta: PT. Elex Media Komputindo.
- H, Nazrudin Safaat. (2012). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android Edisi Revisi*. Bandung: Informatika.
- Handayani Septiaji W. (2012). *Membuat SMS Gateway dengan Delphi 7*. Bandung: Widya Media.
- Kadir Abdul. (2014). *Sistem Informasi*. Bandung: Gudang ilmu.
- Kurniawan, Rulianto. (2008). *Membangun Situs Dengan PHP Untuk Orang Awam*. Palembang: MAXIKOM.
- Komputer Wahana. (2014). *Mudah Membuat Aplikasi SMS Gateway dengan CodeIgniter*. Jakarta: PT. Elex Media Komputindo.
- Ladjamudin Bin Al-Bahra. (2005). *Konsep Diagram Alir Dokumen*. Jakarta.

- Maulana Sofyan. (2015). *5 Proyek Populer SMS Gateway*. Jakarta: PT. Elex Media Komputindo.
- R, N, Mardiani Eri. (2016). *Kumpulan Latihan SQL*. Jakarta: PT. Elex Media Komputindo.
- R, Rante Rintho. (2018). *Pemrograman Web Dasar*. Yogyakarta: CV BUDI UTAMA.
- Romney, Steinbert. (2015). *Analisis dan Desain Sistem Informasi*. Jakarta: Library binus.
- Sidik. (2014). *Xampp Windows Linux*. Yogyakarta: Teknologi Informatika.
- Sidik Betha. (2017). *Pemrograman Web dengan PHP7*. Bandung: Informatika.
- Susanto Heru. (2013). *Bijak Meminjam dan Menggunakan Uang Bank, 101 Kebijakan yang Harus Dimiliki Sebelum Berurusan dengan Bank*. Jakarta: PT. Elex Media Komputindo.
- Sutabri Tata. (2012). *Konsep Sistem Informasi, Edisi ke Satu*. Yogyakarta
- Tri Wicaksono. (2007). *Seri Penuntun Praktis Pemrograman SMS Interaktif Berbasis Java*. Jakarta: PT. Elex Media Komputindo.
- Yuhefizar. (2009). *Cara Mudah Membangun Website Interaktif Menggunakan CMS Joomla (edisi Revisi)*. Jakarta: PT. Elex Media Komputindo.

LAMPIRAN

- **Index.php**

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="">
    <meta name="author" content="Dashboard">
    <title>PT. Jaya Angkasa Globalindo</title>
    <link href="assets/css/bootstrap.css" rel="stylesheet">
    <link href="assets/font-awesome/css/font-awesome.css" rel="stylesheet" />
    <link href="assets/css/style.css" rel="stylesheet">
    <link href="assets/css/style-responsive.css" rel="stylesheet">
    <link rel="shortcut icon" href="">
  </head>
  <div id="login-page">
    <div class="container">
      <div class="login-wrap">
        <form name="form" id="loginF" method="post"
action="" class="form-login">
          <h2 class="form-login-heading">PT. Jaya Angkasa
Globalindo</h2>
          <div id="loading" style="text-align: center"></div>
          <br>
          <div class="form-group">
            <div class="col-sm-12">
              <input type="text" id="username" name="username"
class="form-control" placeholder="Username" autofocus>
            </div>
          </div>
          <br>
          <div class="form-group">
            <div class="col-sm-12">
              <input type="password" id="passlogin" name="passlogin"
class="form-control" placeholder="Password">
            </div>
          </div>
          <div class="form-group">
            <div class="col-sm-12">
              <button class="btn btn-block btn-default"><i class="fa fa-lock"></i>
LOGIN</button>
            </div>
          </div>
        </form>
      </div>
    </div>
  </div>
```

```

</div>
<div class="registration">
<br>
<br>
<br>
<br>
<br>
<br>
<br>
</div>
<footer class="site-footer">
<div class="text-center">
<b>Silahkan Login</b></a>
</div>
</footer>
</form>
</div>
</div>
</div>
<script src="assets/js/jquery.2.1.1.min.js"></script>
<script src="assets/js/bootstrap.min.js"></script>
<script src="assets/js/jquery.validate.min.js"></script>
<script type="text/javascript">
jQuery(function($) {
$.validator.setDefaults({
submitHandler: function () {
login();
}
});
$.ready(function () {
$("#loginF").validate({
errorElement: 'div',
errorClass: 'help-block',
focusInvalid: true,
rules: {
username: {
required: true
},
passlogin: {
required: true
}
},
messages: {
username: "Mohon isi Username anda",
passlogin: "Mohon isi Password anda"
},
highlight: function (e) {

```

```

                $(e).closest('.form-group').removeClass('has-info').addClass('has-
error');
            },
            success: function (e) {
                $(e).closest('.form-group').removeClass('has-error');//.addClass('has-
info');
                $(e).remove();
            }
        })
    });
    function login() {
        $("#loading").html('<div class="alert alert-warning alert-success">Mohon
Tunggu...</div>');
        $.post('cek_login.php', $("form").serialize(), function (hasil) {
            $('form input[type="text"],form input[type="password"]').val("");
            $("#loading").html(hasil);
        });
    }
});
</script>
</body>
</html>

```

- **Media.php**

```

<?php
session_start();

error_reporting(0);

include "config/koneksi.php";

include "config/page.php";

$id_user=$_SESSION['kode'];

$nm_user=$_SESSION['namauser'];

$photo=$_SESSION['photo'];

$sesi_username = isset($_SESSION['username']) ?
$_SESSION['username'] : NULL;

if ($sesi_username != NULL || !empty($sesi_username)
||$_SESSION['leveluser']=='1' )
{

```

```
?>
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="">
    <meta name="author" content="Dashboard">
    <title>PT. Jaya Angkasa Globalindo</title>
    <link href="assets/css/bootstrap.css" rel="stylesheet">
    <link href="assets/font-awesome/css/font-awesome.css" rel="stylesheet" />
    <link rel="stylesheet" type="text/css" href="assets/css/zabuto_calendar.css">
    <link rel="stylesheet" type="text/css" href="assets/css/select2.min.css">
    <link rel="stylesheet" type="text/css" href="assets/lineicons/style.css">
    <link rel="stylesheet" type="text/css" href="<?php echo $ambilcss1; ?>">
    <link rel="stylesheet" type="text/css" href="<?php echo $ambilcss2; ?>">
    <link rel="shortcut icon" href="">
    <link href="assets/css/datepicker.css" rel="stylesheet" type="text/css">
    <link href="assets/css/style.css" rel="stylesheet">
    <link href="assets/css/style-responsive.css" rel="stylesheet">
    <script src="assets/js/chart-master/Chart.js"></script>
    <script src="assets/js/time.js" type="text/javascript"></script>
  </head>
  <body>
    <section id="container" >
      <header class="header black-bg">
        <div class="sidebar-toggle-box">
```

```
<div class="fa fa-bars tooltips" data-placement="right" data-original-
title="Toggle Navigation"></div>
```

```
</div>
```

```
<a href="media.php" class="logo"><b>PT. Jaya Angkasa
Globalindo</b></a>
```

```
<div class="nav notify-row" id="top_menu">
```

```
<ul class="nav top-menu">
```

```
<li class="dropdown" id="header_inbox_bar">
```

```
</li>
```

```
<span style="color:white" id="dates"><span id="the-day"><a>Hari,
00 Bulan 0000</a></span> <span id="the-time"><a>00:00:00</a></span>
</span>
```

```
</ul>
```

```
</div>
```

```
<div class="top-menu">
```

```
<ul class="nav pull-right top-menu">
```

```
<li><a class="logout" href="logout.php" onclick="return
confirm('Apakah anda yakin ?');"><i class="glyphicon glyphicon-
off"></i><span> Logout</span></a></li>
```

```
</ul>
```

```
</div>
```

```
</header>
```

```
<aside>
```

```
<div id="sidebar" class="nav-collapse ">
```

```
<ul class="sidebar-menu" id="nav-accordion">
```

```
<li class="mt">
```

```
<a class="<?php echo $classmenu1 ?>" href="?id=home">
```

```
<i class="fa fa-dashboard"></i>
```

```
<span>Dashboard</span>
```



```

        </a>
    </li>
    <li class="sub-menu">
        <a href="javascript:;" >
            <i class="fa fa-envelope"></i>
            <span>SMS</span>
        </a>
        <ul class="sub active">
            <li><a data-toggle="modal" data-target="#myModal" href=""
>Tulis Pesan</a></li>
            <li><a class="<?php echo $classmenu2 ?>"
href="?id=inbox">Kotak Masuk</a></li>
            <li><a class="<?php echo $classmenu3 ?>"
href="?id=outbox">Kotak Keluar</a></li>
            <li><a class="<?php echo $classmenu4 ?>"
href="?id=sent">Kontak Terkirim</a></li>
            <li><a data-toggle="modal" data-target="#modalsiaran"
href="">Pesan Tersiar</a></li>
        </li>
    </ul>
</li>
<li class="active sub-menu">
    <a href="javascript:;" >
        <i class="fa fa-book"></i>
        <span>Kontak</span>
    </a>
    <ul class="sub">
        <li><a class="<?php echo $classmenu8 ?>"
href="?id=pb">Daftar kontak</a></li>

```

```
        <li><a class="<?php echo $classmenu9 ?>"
href="?id=grp">Daftar Group</a></li>
    </ul>
</li>
<li class="sub-menu">
    <a href="javascript:;" >
        <i class="fa fa-folder-o"></i>
        <span>Auto SMS</span>
    </a>
    <ul class="sub active">
        <li><a class="<?php echo $classmenu5 ?>"
href="?id=schedule">Pesan Terjadwal</a></li>
    </ul>
</li>
<li class="mt">
    <a class="logout" href="logout.php" onclick="return
confirm('Apakah anda yakin ?');">
        <i class="glyphicon glyphicon-off"></i>
        <span>logout</span>
    </a>
</li>
</ul>
</div>
</aside>

<section id="main-content">
    <section class="wrapper">
        <?php
```

```
        include "modal.php";
        include "config/koneksi.php";
        include "action.php";
        ?>
        <?php include $ambil; ?>
    </section>

    <script src="assets/js/jquery-1.8.3.min.js"></script>
    <script src="assets/js/bootstrap.min.js"></script>
    <script class="include" type="text/javascript"
src="assets/js/jquery.dcjaccordion.2.7.js"></script>
    <script src="assets/js/jquery.scrollTo.min.js"></script>
    <script src="assets/js/jquery.nicescroll.js" type="text/javascript"></script>
    <script src="assets/js/jquery.sparkline.js"></script>
    <script src="assets/js/common-scripts.js"></script>
    <script src="assets/js/select2.min.js"></script>
    <script src="assets/js/bootstrap-datepicker.min.js"></script>
    <script src="<?php echo $ambiljs1; ?>"></script>
    <script src="<?php echo $ambiljs2; ?>"></script>
    <script src="<?php echo $ambiljs3; ?>"></script>
    <script src="<?php echo $ambiljs4; ?>"></script>
    <?php include $ambilfungsi; ?>
    <script type="text/javascript">
        $("#tgl_lahir").datepicker({
            autoclose: true,
            todayHighlight: true
        });
        function Ajax()
```

```

{
    var
        $http,
        $self = arguments.callee;
    if (window.XMLHttpRequest) {
        $http = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        try {
            $http = new ActiveXObject('Msxml2.XMLHTTP');
        } catch(e) {
            $http = new ActiveXObject('Microsoft.XMLHTTP');
        }
    }
    if ($http) {
        $http.onreadystatechange = function()
        {
            if (/4|^complete$/.test($http.readyState)) {
                document.getElementById('header_inbox_bar').innerHTML =
                $http.responseText;
                setTimeout(function(){ $self(); }, 10000);
            }
        };
        $http.open('GET', 'cek_inbox.php' + '?' + new Date().getTime(), true);
        $http.send(null);
    }
    else {
        document.getElementById('header_inbox_bar').innerHTML =
        $http.responseText;
    }
}

```

```
    }  
  }  
</script>  
<script type="text/javascript">  
  setTimeout(function() {Ajax();}, 10000);  
  $(document).ready(function(){  
    $("#nohp").select2({  
      ajax: {  
        url: 'cariphone.php',  
        dataType: 'json',  
        delay: 250,  
        processResults: function (data) {  
          return {  
            results: data  
          };  
        },  
        cache: true  
      }  
    });  
    $("#group").select2({  
      ajax: {  
        url: 'carigroup.php',  
        dataType: 'json',  
        delay: 250,  
        processResults: function (data) {  
          return {  
            results: data  
          }  
        }  
      }  
    });  
  });  
</script>
```

```

        };
    },
    cache: true
}
});
});
</script>
</body>
</html>
<?php
}else{
    session_destroy();
    header('Location:index.php?status=Silahkan Login');
}
?>

```

- **Logout.php**

```

<?php session_start();
if(isset($_SESSION['username']))
{
    session_destroy();
    header('Location:index.php?status=Anda sudah Keluar');
}else{
    session_destroy();
    header('Location:index.php?status=Silahkan Login!');
}
?>

```

