

**PEMBUATAN SISTEM PEMESANAN TRAVEL DAN
PENGIRIMAN BARANG MENGGUNAKAN TRAVEL
DENGAN LAYANAN WAP**

SKRIPSI

*Diajukan untuk memenuhi sebagian persyaratan untuk
memperoleh gelar Sarjana Teknik*

UNIVERSITAS BRAWIJAYA



Disusun oleh :

**Candra Handi Aribowo
NIM. 0210633020-63**

**DEPARTEMEN PENDIDIKAN NASIONAL
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2009**

**PEMBUATAN SISTEM PEMESANAN TRAVEL DAN
PENGIRIMAN BARANG MENGGUNAKAN TRAVEL
DENGAN LAYANAN WAP**

SKRIPSI

UNIVERSITAS BRAWIJAYA

Disusun oleh :

**CANDRA HANDI ARIBOWO
NIM. 0210633020**

Telah diperiksa dan disetujui oleh
Dosen Pembimbing :

Arief Andy Soebroto ST., M.Kom.
NIP. 19720425 199903 1 002

Himawat Aryadita ST., M.Sc
NIP. 19801018 200801 1 003

**PEMBUATAN SISTEM PEMESANAN TRAVEL DAN
PENGIRIMAN BARANG MENGGUNAKAN TRAVEL
DENGAN LAYANAN WAP**

Disusun oleh :

**Candra Handi Aribowo
NIM. 0210633020-63**

Skripsi ini telah diuji dan dinyatakan lulus pada
Tanggal 12 bulan Agustus 2009

DOSEN PENGUJI

**Ir. Heru Nurwasito, M.Kom
NIP. 19650402 199002 1 001**

**Suprpto ST.,MT
NIP. 19710727 199603 1 001**

**Adharul Muttaqin ST.,MT
NIP. 19760121 200501 1 001**

**Mengetahui
Ketua Jurusan Teknik Elektro**

**Ir. Heru Nurwasito, M.Kom.
NIP. 19650402 199002 1 001**

KATA PENGANTAR

Puji Syukur kehadiran Allah SWT karena dengan berkat rahmat dan karunia serta ridlo-Nya penyusunan skripsi ini dengan judul “Pembuatan Sistem Pemesanan Travel dan Pengiriman Barang Menggunakan Travel Dengan Layanan WAP ” dapat diselesaikan. Penulis menyadari bahwa kajian ini tidak akan mencapai titik akhir penyelesaian tanpa bantuan berbagai pihak, karenanya penulis mengucapkan terima kasih kepada :

1. Tuhan Yang Maha Esa Penguasa alam semesta.
2. Ibunda dan Ayahanda tercinta, Djajadi dan Suci Sri Wulandari terimakasih untuk seluruh doa dan dukungannya yang telah diberikan kepada Ananda selama studi hingga terselesaikannya skripsi ini.
3. Adik-adikku yang tercinta, Agnesia Striata S.C.P dan Delaneira H.K.C, terima kasih atas gangguan-gangguan serta doa yang diberikan kepada ananda selama ini.
4. Ir. Heru Nurwarsito, M.Kom. selaku Ketua Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya.
5. Rudy Yuwono, ST, MSc. selaku Sekretaris Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya.
6. Bapak Arief Andy Soebroto ST.,M.Kom. selaku dosen pembimbing I pada penyusunan skripsi ini.
7. Bapak Himawat Aryadita ST.,MT.,MSc selaku dosen pembimbing II pada penyusunan skripsi ini.
8. Bapak dan Ibu dosen serta karyawan Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya.
9. Temanku zidni, angga, febr, samid, farid, bograh, ebeb, pak eko, samjuls, ajis, eriek, didik, erga dan sahabat-sahabatku yang tidak dapat aku sebutkan satu-persatu, yang telah bersama menjalani hari dengan suka dan duka. Terima kasih atas pelajaran hidup yang sangat berharga, semua kenangan tidak akan bisa terlupakan.

10. Serta pihak yang telah turut membantu baik secara langsung, mas gembong, daging dan mas salis, maupun pihak-pihak yang secara tidak langsung ikut membantu dalam penyelesaian skripsi ini.

Tiada gading yang tak retak, tersadar bahwa skripsi ini sangat jauh dari kesempurnaan. Karenanya, segala kritik dan saran yang sifatnya membangun dari pembaca tentang isi skripsi ini akan diterima dengan senang hati. Akhir kata, penulis berharap, semoga skripsi ini dapat bermanfaat bagi semua pihak yang membutuhkan.

Malang, Agustus 2009,

Penyusun



ABSTRAK

Candra Handi Aribowo. 2009. PEMBUATAN SISTEM PEMESANAN JASA DAN PENGIRIMAN BARANG MENGGUNAKAN TRAVEL DENGAN LAYANAN WAP. Dosen Pembimbing: Arief Andy Soebroto, ST., MKom dan Himawat Aryadita, ST., MT., M.Sc

WAP adalah sebuah protokol yang menghubungkan internet dengan handphone. Pengguna handphone dengan layanan WAP dapat mengakses informasi dan bertransaksi di Internet langsung melalui handphone sebagaimana akses Internet dengan komputer biasa. *Mobile Information WAP* adalah suatu sistem informasi bergerak yang bertujuan untuk memberi kemudahan seseorang yang ingin mendapatkan suatu informasi tanpa dibatasi waktu, ruang serta letak geografis. Keuntungan bagi penyedia jasa travel dilihat dari segi perebutan pangsa pasar karena dengan bervariasinya cara pemesanan suatu jasa, maka akan dapat menjangkau semua lapisan masyarakat dan kemudahan dalam mendapatkan suatu pelayanan akan berdampak positif terhadap persaingan usaha. Keuntungan bagi pemakai jasa travel dengan adanya sistem pemesanan dengan layanan WAP ini adalah pemakai jasa dapat menggunakan sistem ini sebagai alternatif pemesanan jasa yang diinginkannya kapan saja dan dimana saja selama terdapat jaringan GPRS dan handphone.

Sistem Pemesanan Jasa Travel dengan layanan WAP ini dikembangkan melalui beberapa tahap, yaitu: analisis, desain, implementasi dan pengujian. Sistem Pemesanan Jasa Travel dengan layanan WAP terdiri dari sebuah wap portal yang dapat diakses dengan handphone dengan menu statik jasa travel, yaitu : pemesanan travel dan kurir, pembatalan pemesanan, lihat info transaksi, daftar anggota dan profil penyedia jasa travel. Selain sebuah wap portal, juga terdapat sebuah web portal yang dapat diakses oleh pihak manajemen untuk melihat laporan bulanan, mengisi data yang berhubungan dengan travel baik. Sistem Pemesanan Jasa Travel dengan layanan WAP dikembangkan dengan menggunakan bahasa pemrograman PHP untuk *websitenya*, bahasa pemrograman WML untuk *wapsitenya*, MySQL sebagai databasenya dan web server Apache.

Kata Kunci : Travel, Website, WAP, WML, Handphone, GPRS.

DAFTAR ISI

LEMBAR PERSETUJUAN.....	i
KATA PENGANTAR.....	iii
ABSTRAK.....	v
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	4
1.5 Manfaat.....	4
1.6 Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 <i>Global System for Mobile Communication (GSM)</i>	8
2.1.1 Teknologi Tanpa Kabel untuk teknologi selular.....	10
2.2 <i>General Packet Radio Service (GPRS)</i>	11
2.2.1 <i>Wireless Application Protocol (WAP)</i>	14
2.2.2 <i>Wireless Markup Language (WML)</i>	16
2.2.2.1 Elemen <i>Deck</i> dan <i>card</i>	18
2.2.2.2 Elemen <i>event</i>	18
2.2.2.3 Elemen <i>Task</i>	19
2.2.2.4 Elemen Variabel.....	20
2.2.2.5 Elemen <i>User Input</i>	20
2.2.2.6 Elemen <i>Anchor</i>	20
2.2.2.7 Elemen <i>Image</i>	20
2.2.2.8 Elemen <i>Format Text</i>	20
2.3 PHP.....	22
2.4 <i>Apache Web Server</i>	24
2.5 MySQL.....	24
BAB III METODE PENELITIAN.....	26
3.1 Studi Literatur.....	26
3.2 Analisis dan Perancangan.....	26
3.3 Implementasi.....	27
3.4 Pengujian dan Analisis.....	27

3.5 Pengambilan Kesimpulan dan Saran.....	26
BAB IV PERANCANGAN.....	27
4.1 Analisis Proses Bisnis.....	28
4.1.1 Analisis Proses Bisnis Travel.....	28
4.1.2 <i>Standart Operational Procedure</i>	28
4.1.2.1 Sistem Pemesanan Jasa Travel Secara Manual	29
4.1.2.2 Sistem Pemesanan Jasa Travel Menggunakan WAP.....	31
4.1.3 Diagram Alir Sistem Pemesanan Travel	32
4.1.3.1 Perbedaan Sistem Pemesanan Travel Secara Manual dan WAP.....	33
4.1.4 Analisis Kebutuhan Perangkat Lunak.....	34
4.1.4.1 Spesifikasi Pengguna Perangkat Lunak.....	36
4.1.4.2 Spesifikasi Perangkat Lunak	38
4.2 Perancangan Perangkat Lunak.....	40
4.2.1 Diagram Blok Sistem	40
4.2.2 Data Flow Diagram	42
4.2.2.1 <i>Context</i> Diagram	42
4.2.2.2 DFD Level 1	46
4.2.2.3 DFD Level 2	53
4.2.2.3.1 DFD Level 2 Proses WAP Pemesanan Jasa	53
4.2.2.3.2 DFD Level 2 Proses WEB Pemesanan Jasa	57
4.2.2.3 DFD Level 3	62
4.2.2.3.1 DFD/Level 3 Validasi Login Anggota.....	62
4.2.2.3.2 DFD Level 3 Daftar	63
4.2.2.3.3 DFD Level 3 Olah Data Pemesanan	64
4.2.2.3.4 DFD Level 3 Validasi Login Admin, CS & Manajemen	69
4.2.2.3.5 DFD Level 3 Olah Data Manajemen	71
4.2.2.3.5 DFD Level 3 Olah Data Customer Service	73
4.2.2.3.5 DFD Level 3 Olah Data Admin	74
4.2.3 Perancangan Basis Data	76
4.2.3.1 ERD (<i>Entity Relationship Diagram</i>).....	76
4.2.3.2 Normalisasi Data	78



4.2.3.3	<i>Data Object Description</i>	79
4.2.3.3.1	<i>Data Type Definition</i>	79
4.2.4	<i>Site Map Sistem</i>	82
4.2.5	<i>Perancangan Antarmuka User</i>	83
4.2.5.1	<i>Antarmuka User Pengunjung dan Anggota</i>	83
4.2.5.1.1	<i>Tampilan Index Pengunjung dan Anggota</i>	83
4.2.5.1.2	<i>Tampilan Registrasi Anggota</i>	84
4.2.5.1.3	<i>Tampilan Login Anggota</i>	84
4.2.5.1.4	<i>Tampilan Konfirmasi Login Anggota</i>	85
4.2.5.1.5	<i>Tampilan Informasi Jasa Travel</i>	86
4.2.5.1.6	<i>Tampilan Informasi Perusahaan</i>	86
4.2.5.1.7	<i>Tampilan Informasi Transaksi</i>	87
4.2.5.2	<i>Antarmuka User Customer Service</i>	87
4.2.5.2.1	<i>Tampilan Login admin dan manajemen</i>	87
4.2.5.2.2	<i>Tampilan menu utama customer service</i>	87
4.2.5.3	<i>Antarmuka User Manajemen</i>	88
4.2.5.3.1	<i>Tampilan Menu Utama Manajemen</i>	88
4.2.5.4	<i>Antarmuka User admin</i>	89
4.2.5.4.1	<i>Tampilan Menu Utama Admin</i>	89
BAB V IMPLEMENTASI PERANGKAT LUNAK		90
5.1	<i>Implementasi Sistem</i>	90
5.1.1	<i>Spesifikasi Hardware</i>	91
5.1.2	<i>Spesifikasi Software</i>	91
5.2	<i>Implementasi Algoritma</i>	91
5.2.1	<i>Algoritma Koneksi</i>	91
5.2.2	<i>Algoritma Session</i>	93
5.2.3	<i>Algoritma Log In</i>	94
5.2.4	<i>Algoritma Input Data</i>	95
5.2.5	<i>Algoritma Hapus Data</i>	96
5.2.6	<i>Algoritma Transaksi Pemesanan</i>	98
5.2.7	<i>Algoritma Transaksi Pembatalan</i>	99
5.3	<i>Implementasi Antarmuka</i>	101
5.3.1	<i>Implementasi Antarmuka Halaman WAP Pengunjung</i>	101
5.3.2	<i>Implementasi Antarmuka Login</i>	102



5.3.3	Implementasi Registrasi Anggota	103
5.3.4	Implementasi Jadwal Keberangkatan	103
5.3.5	Implementasi Halaman Menu Anggota.....	104
5.3.6	Implementasi Halaman Konfirmasi Pesanan.....	105
5.3.7	Implementasi Halaman Detail Pesanan	105
5.3.8	Implementasi Antarmuka WEB user admin dan manajemen.....	106
5.3.9	Implementasi Antarmuka Manajemen	107
5.3.10	Implementasi Antarmuka Admin	108
5.3.11	Implementasi Antarmuka Edit Data	108
5.3.12	Implementasi Antarmuka Edit Data Anggota	109
5.3.13	Implementasi Antarmuka Lihat Laporan.....	109
BAB VI PENGUJIAN DAN ANALISIS.....		110
6.1	Pengujian Algoritma.....	110
6.1.1	Pengujian Proses Login.....	110
6.1.2	Pengujian Proses Tambah Data	112
6.1.3	Pengujian Proses Edit Data.....	113
6.1.4	Pengujian Proses Hapus Data	115
6.2	Pengujian Validasi.....	117
6.2.1	Pengujian Sisi WAP.....	117
6.2.2	Pengujian Sisi WEB.....	122
6.2.3	Hasil Pengujian Validasi	129
6.3	Pengujian Query	132
BAB VII.....		136
PENUTUP.....		136
7.1	Kesimpulan.....	136
7.2	Saran.....	136
LAMPIRAN		137
DAFTAR PUSTAKA.....		138

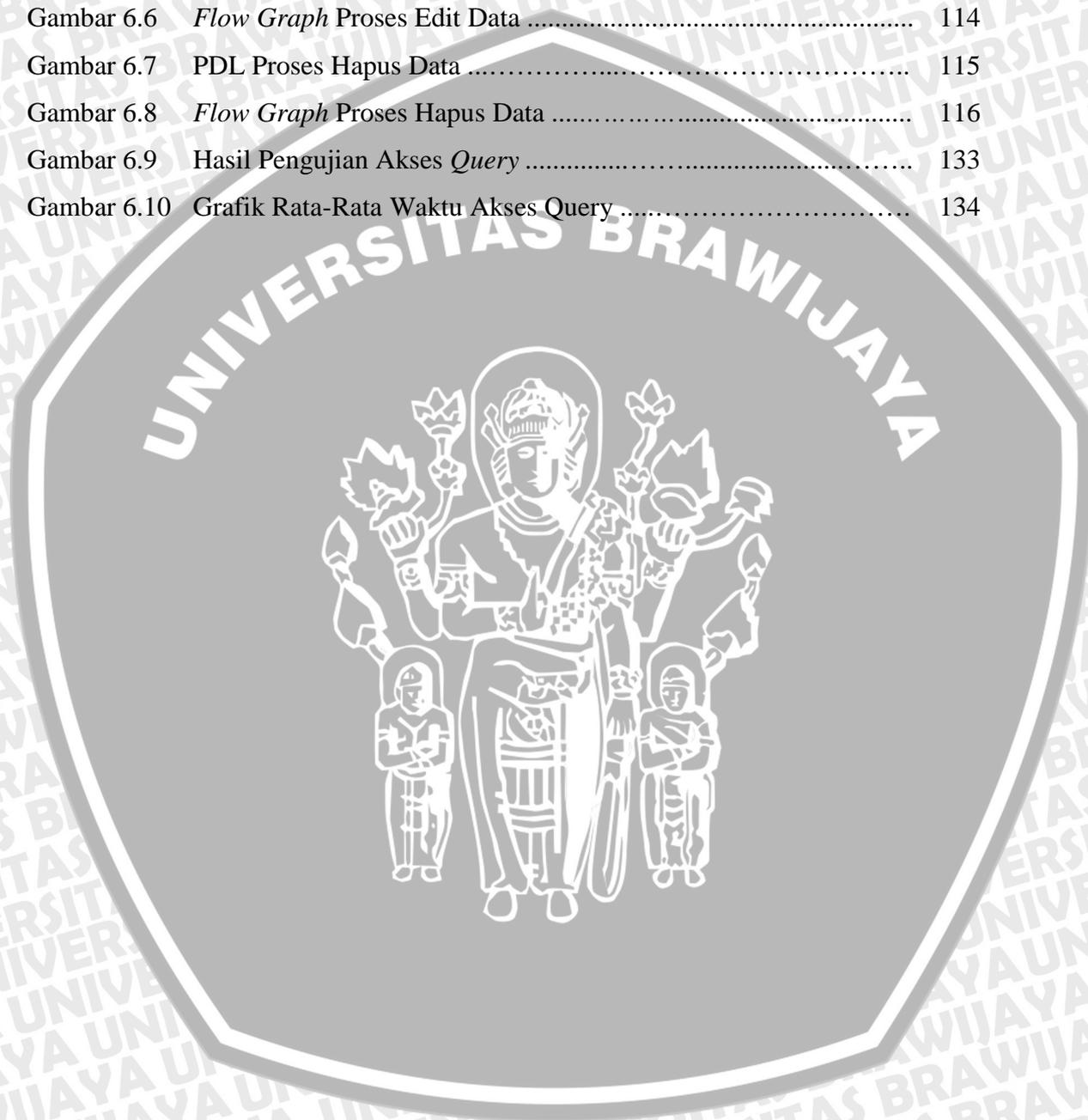
DAFTAR GAMBAR

Gambar 2.4.	GSM Network	10
Gambar 2.5.	Arsitektur GPRS	13
Gambar 2.6.	Layar Protokol WAP	15
Gambar 2.7	Skema Dasar PHP.....	23
Gambar 4.1	Diagram Pohon Perancangan Sistem	27
Gambar 4.2	Diagram Alir Pemesanan Travel Secara Manual	33
Gambar 4.3	Diagram Alir Pemesanan Travel Dengan Layanan WAP	34
Gambar 4.4	Diagram Blok Sistem Aplikasi Sistem Pemesanan Jasa Travel Secara Online	41
Gambar 4.5	Context Diagram Aplikasi Sistem Pemesanan Jasa Travel	42
Gambar 4.6	DFD level 1 Sistem Pemesanan Jasa Travel Dengan Layanan WAP	47
Gambar 4.7	DFD level 2 Proses WAP Pemesanan Travel	54
Gambar 4.8	DFD level 2 Proses Web Pemesanan Travel	58
Gambar 4.9	DFD level 3 Proses Validasi Login Anggota	62
Gambar 4.10	DFD level 3 Proses Daftar	63
Gambar 4.11	DFD level 3 Olah Data Pemesanan	65
Gambar 4.12	DFD level 3 Proses Validasi Login Admin & Manajemen	68
Gambar 4.13	DFD level 3 Proses Olah Data Manajemen	71
Gambar 4.14	DFD level 3 Proses Olah Data Admin	73
Gambar 4.15	DFD level 3 Proses Olah Data Admin Perangkat Lunak	75
Gambar 4.16	ER Diagram Sistem Pemesanan Jasa Travel	77
Gambar 4.17	Normalisasi Tabel dalam Basis Data Sistem Pemesanan Travel .	78
Gambar 4.18	Sitemap Sistem Pemesanan Jasa Travel (Antarmuka WAP)	82
Gambar 4.19	Sitemap Sistem Pemesanan Jasa Travel (Antarmuka WEB).....	83
Gambar 4.20	Halaman Index Sistem Pemesanan Jasa Travel (WAP)	84
Gambar 4.21	Halaman Daftar Anggota Baru Pengunjung	84
Gambar 4.22	Halaman Login Anggota	85
Gambar 4.23	Halaman Konfirmasi Login	85
Gambar 4.24	Halaman Kedua Informasi Jasa Travel	86
Gambar 4.25	Halaman Profil Perusahaan	86
Gambar 4.26	Halaman Pertama Informasi Transaksi	87



Gambar 4.27	Halaman Kedua Informasi Transaksi	87
Gambar 4.28	Halaman Login Admin, Menu Utama dan Manajemen	88
Gambar 4.29	Halaman Masukkan dan Edit Data	88
Gambar 4.30	Halaman Lihat Laporan	89
Gambar 4.31	Halaman Menu Utama Admin	89
Gambar 5.1	Diagram Pohon Implementasi Sistem	90
Gambar 5.2	Algoritma Koneksi	92
Gambar 5.3	Algoritma Proses Koneksi	92
Gambar 5.4	Algoritma Cek <i>Session</i> Untuk Anggota	93
Gambar 5.5	Algoritma Proses <i>Session</i>	93
Gambar 5.6	Algoritma Validasi <i>Login</i>	94
Gambar 5.7	Algoritma Proses Login	95
Gambar 5.8	Algoritma <i>Input Data</i>	95
Gambar 5.9	Algoritma Proses Input Data	96
Gambar 5.10	Algoritma Hapus Data	97
Gambar 5.11	Algoritma Proses Hapus Data	97
Gambar 5.12	Algoritma Transaksi Pemesanan	98
Gambar 5.13	Algoritma Proses Transaksi Pemesanan	99
Gambar 5.14	Algoritma Transaksi Pembatalan Tiket	100
Gambar 5.15	Algoritma Proses Pembatalan Transaksi	101
Gambar 5.16	Implementasi Halaman Utama (WAP)	102
Gambar 5.17	Implementasi Halaman Login Anggota	102
Gambar 5.18	Implementasi Halaman <i>Registrasi</i> Anggota	103
Gambar 5.19	Implementasi Halaman Jadwal Keberangkatan	104
Gambar 5.20	Implementasi Halaman Menu Anggota	104
Gambar 5.21	Implementasi Halaman Konfirmasi Pemesanan	105
Gambar 5.22	Implementasi Halaman Detail Pesanan	105
Gambar 5.23	Implementasi Fasilitas <i>Login User</i> Admin dan Manajemen	106
Gambar 5.24	Halaman Edit dan Input Data <i>Customer Service</i>	107
Gambar 5.25	Halaman Antarmuka Manajemen	107
Gambar 5.26	Halaman Antarmuka Admin	108
Gambar 5.27	Halaman Edit Data Travel ..	108
Gambar 5.28	Halaman Edit Data Anggota	109
Gambar 5.29	Halaman Lihat Laporan Data Anggota	109

Gambar 6.1	PDL Proses Login	110
Gambar 6.2	<i>Flow Graph</i> Proses Login	111
Gambar 6.3	PDL Proses Tambah Data	112
Gambar 6.4	<i>Flow Graph</i> Proses Tambah Data	112
Gambar 6.5	PDL Proses Edit Data	114
Gambar 6.6	<i>Flow Graph</i> Proses Edit Data	114
Gambar 6.7	PDL Proses Hapus Data	115
Gambar 6.8	<i>Flow Graph</i> Proses Hapus Data	116
Gambar 6.9	Hasil Pengujian Akses <i>Query</i>	133
Gambar 6.10	Grafik Rata-Rata Waktu Akses <i>Query</i>	134



BAB I

PENDAHULUAN

1.1 Latar Belakang

Dewasa ini kebutuhan masyarakat akan kecepatan akses semakin meningkat. Bukti bahwa masyarakat membutuhkan kecepatan akses misalnya adalah keinginan dari masyarakat untuk mendapatkan tiket pesawat terbang tanpa harus mengantri. Masyarakat tentunya ingin secara cepat mendapatkan informasi seputar jadwal pemberangkatan, pemesanan tempat duduk, sampai pemesanan tiket, tanpa harus datang secara fisik. Kesimpulannya bahwa saat ini masyarakat memerlukan suatu teknologi untuk mempermudah dalam memperoleh fasilitas-fasilitas yang diinginkan khususnya fasilitas dalam jasa perjalanan.

Jasa Travel merupakan salah satu sarana transportasi yang cukup diminati untuk perjalanan antar kota dalam propinsi. Cara pemesanan Jasa Travel baik dalam hal jasa perjalanan maupun jasa pengiriman relatif cukup menyita waktu karena pemesanan harus dilakukan secara manual dan juga pendataannya dilakukan secara manual oleh pihak manajemen. Karena itu perlu dibuat sebuah alternatif pemesanan Jasa Travel. Salah satu alternatif yang dikemukakan adalah pemesanan Jasa Travel menggunakan jaringan GPRS (*Global Packet Radio Service*) dan teknologi WAP (*Wireless Application Protocol*).

Wireless Application Protocol (WAP) merupakan langkah awal menuju mobile internet, yang memungkinkan sebuah ponsel bisa mengakses internet. WAP adalah sebuah standar komunikasi antara *mobile telephone* dengan informasi yang ada dalam internet [TEL-04 : 02-03].

Konsep WAP adalah menggabungkan dua industri dan teknologi yang sedang berkembang pesat, yakni wireless dan internet. Mengapa mesti handphone yang lebih potensial dipilih untuk aplikasi WAP ? Alasan yang paling sederhana adalah karena handphone adalah media komunikasi yang mudah dibawa-bawa ke mana-mana [TEL-04 : 02-03]. Para perancang WAP berharap agar informasi yang ada di internet mudah diakses oleh siapa saja dan dimana saja asalkan memiliki handphone dengan aplikasi WAP dan terdapat jaringan GPRS.

Penerapan *Wireless Application Protocol* diharapkan akan memberi kemudahan kepada pelanggan untuk tetap dapat memperoleh informasi serta melakukan transaksi yang berhubungan dengan jasa perjalanan. Sistem layanan *mobile* untuk jasa ini adalah sebuah revolusi sistem pelayanan kepada pelanggan dengan memanfaatkan teknologi internet yang hadir dalam PC sehingga mereka bisa melakukan aktivitas pemesanan tiket dan sebagainya kapanpun dan dimanapun mereka berada [TEL-04 : 02].

Perancangan dan pembuatan perangkat lunak ini diharapkan mampu memberikan keuntungan kepada kedua belah pihak, baik dari pihak pelanggan maupun dari pihak pengelola jasa travel. Perangkat lunak ini merupakan pengaplikasian dari titik temu antara kebutuhan pelanggan dengan kebutuhan jasa Travel.

Sistem ini memiliki keunggulan dapat diakses dimana saja dan kapan saja. Protokol WAP digunakan sebagai protokol yang mengatur pertukaran informasi antara user dengan server. Protokol WAP juga digunakan untuk menerjemahkan kode WML (Wireless Markup Language) sehingga bisa ditampilkan pada layar telepon genggam. Jaringan GPRS berguna sebagai jaringan pembawa data yang ditransmisikan dari user menuju server dan begitu pula sebaliknya. Kendala yang akan ditemui adalah kelancaran koneksi antara server WAP dengan telepon genggam. Kelancaran koneksi tersebut tergantung jenis telepon genggam dan kondisi traffic jaringan GPRS operator yang digunakan.

Sistem pemesanan Jasa Travel melalui jaringan GPRS menggunakan Layanan WAP ini akan dibuat menggunakan bahasa pemrograman PHP dan WML. PHP digunakan sebagai software untuk membaca data pada database dan menangani permintaan yang dikirimkan oleh user. WML digunakan untuk merancang tampilan dari aplikasi ini.

1.2 Rumusan Masalah

Permasalahan yang diangkat pada Tugas Akhir ini adalah:

1. Merancang dan mengimplementasikan Sistem Pemesanan Travel dan Pengiriman Barang dengan Travel menggunakan WAP.
2. Merancang aplikasi untuk melakukan proses input data, update data, hapus data, informasi jadwal pemberangkatan, transaksi, laporan transaksi dan segala fitur yang berhubungan dengan sistem yang akan dibuat.
3. Mengimplementasikan aplikasi Sistem Pemesanan Travel yang mengacu pada perancangan aplikasi. Implementasi aplikasi sistem dilakukan pada spesifikasi

lingkungan yang didapatkan dari analisis kebutuhan dan proses perancangan perangkat lunak.

4. Menguji aplikasi Sistem Pemesanan Travel dan Pengiriman Barang meliputi sistem pemesanan travel, sistem pembayaran, sistem pengiriman barang dan segala fitur yang berhubungan dengan manajemen sistem pemesanan dan pengiriman barang menggunakan travel.

1.3 Batasan Masalah

Perlu dilakukan beberapa pembatasan permasalahan dengan tujuan agar pembahasan yang dilakukan tidak meyimpang dari tujuan. Batasan masalah dari sistem yang dirancang adalah :

1. Membuat aplikasi untuk mengakses menu-menu yang ada dalam sistem informasi jasa travel, antara lain : jadwal pemberangkatan, pemesanan travel, pembatalan pemesanan travel, jenis pemesanan travel, jenis perjalanan, jumlah penumpang, biaya untuk pengantaran penumpang, biaya untuk pengantaran barang.
2. *Server* yang selalu terkoneksi dengan internet.
3. Menggunakan *handphone* yang terdapat layanan WAP serta menggunakan jaringan GPRS.
4. Bahasa pemrograman yang digunakan PHP dan WML, untuk database menggunakan MySQL.
5. Belum menangani masalah pengamanan sistem.
6. Pembahasan proses pembayaran difokuskan pada pembayaran antara pihak pemesan / konsumen dengan pihak pengelola travel.
7. Aplikasi perangkat lunak ini dibuat dengan menggunakan teknologi *Wireless Application Protocol* yang pengujiannya menggunakan emulator dan *handphone* yang dilengkapi dengan fasilitas WAP serta memiliki jaringan GPRS.
8. Proses pembayaran atas tiket yang telah dipesan adalah dilakukan dengan menggunakan transfer rekening bank dari pihak pemesan ke rekening bank dari pihak travel yang bersangkutan atau dengan cara COD (Cash On Delivery).
9. Pembatalan pemesanan Jasa Travel bagi calon pelanggan dapat dilakukan sebelum jadwal pembatalan yang sudah ditentukan. Jika proses pembatalan melebihi jangka waktu pembatalan yang sudah ditentukan sebelumnya maka akan dikenakan aturan pemotongan.

10. Bukti bahwa calon pelanggan telah melakukan pemesanan Jasa Travel baik jasa perjalanan maupun jasa pengiriman barang adalah berupa nomer transaksi, yaitu nomer pemesanan Jasa Travel

1.4 Tujuan

Tujuan dari Tugas Akhir ini adalah membuat aplikasi *wireless* pada pemesanan Jasa Travel baik pemesanan jasa antar maupun pengiriman barang sehingga tersedia berbagai macam fasilitas layanan informasi dan transaksi kepada pelanggan yang dapat diakses melalui *mobile device* seperti pocket PC, PDA, dan mobile phone dengan memanfaatkan teknologi *Wireless Application Protocol*.

Aplikasi ini diharapkan dapat mempermudah pengguna jasa travel dalam mengakses informasi jadwal pemberangkatan travel, melakukan transaksi pemesanan Jasa Travel tanpa harus datang secara fisik, dan informasi tentang tarif Jasa Travel baik tarif jasa antar maupun tarif jasa pengiriman barang, sehingga menghemat dari segi biaya dan waktu.

1.5 Manfaat

Adapun manfaat dari penyusunan tugas akhir ini adalah :

1. Bagi Penyusun

Mengembangkan ilmu pengetahuan yang telah diperoleh di bangku kuliah dan mempraktekkannya untuk menyelesaikan permasalahan yang timbul dalam keadaan yang sebenarnya.

2. Bagi Masyarakat

Memberikan alternatif untuk mendapatkan informasi maupun pemesanan Jasa Travel baik jasa perjalanan maupun jasa pengiriman barang.

1.6 Sistematika Penulisan

Pembahasan Tugas Akhir ini terdiri dari 7 bab yang disusun dengan sistematika penulisan sebagai berikut :

Bab I . Pendahuluan

Memuat latar belakang, rumusan masalah, batasan masalah, tujuan serta sistematika penulisan

Bab II. Dasar Teori

Membahas teori-teori yang menunjang dalam pembuatan program aplikasi ini.

Bab III. Metodologi

Membahas metodologi yang digunakan dalam penulisan yang terdiri dari studi literature, perancangan, implementasi, pengujian dan analisa, serta pengambilan kesimpulan dan saran.

Bab IV. Perancangan

Membahas perancangan program yang sesuai dengan teori yang ada

Bab V. Implementasi

Membahas lingkungan implementasi, batasan implementasi, file-file implementasi, algoritma operasi yang diimplementasikan

Bab VI. Pengujian dan Analisa

Membahas pengujian dan analisa program yang dibuat

Bab VII. Penutup

Berisi kesimpulan-kesimpulan yang dapat diambil oleh penulis dalam perancangan aplikasi sistem yang dibuat serta sarana yang menurut penulis sangat diperlukan untuk pengembangan aplikasi ini pada masa yang akan datang

BAB II

TINJAUAN PUSTAKA

Pada bab ini membahas kajian pustaka dan dasar teori yang digunakan untuk penulisan skripsi dan pengembangan Aplikasi Sistem Pemesanan Travel dan Pengiriman Barang menggunakan Travel dengan layanan WAP. Kajian pustaka yang dipaparkan adalah mengenai penerapan teknologi WAP melalui jaringan GPRS sebagai pertukaran data pada sistem pemesanan travel.

Untuk menunjang penulisan skripsi dan pengembangan perangkat lunak sistem pemesanan jasa travel dan jasa pengiriman barang menggunakan travel ini, dibutuhkan kajian terhadap dasar teori yang relevan. Beberapa dasar teori yang dimaksud diantaranya adalah teori tentang :

1. *Global System for Mobile Communication (GSM).*
2. *General Packet Radio Service (GPRS).*
3. *Wireless Application Protocol (WAP).*
4. *Wireless Markup Language (WML).*
5. PHP.
6. *Apache Web Server.*
7. MySQL.

Teori-teori ini akan digunakan untuk mengembangkan (*develop*) sistem tersebut.

2.1 Penerapan layanan WAP menggunakan Jaringan GPRS pada sistem pemesanan travel

WAP adalah sebuah protokol yang menghubungkan internet dengan handphone. Pengguna handphone dengan menggunakan WAP dapat mengakses informasi dan bertransaksi di Internet langsung dengan handphone sebagaimana akses Internet dengan komputer biasa, akses dengan WAP ini juga memerlukan keamanan tinggi. Pemrograman WAP sendiri terdiri dari tiga bagian (fungsi) yaitu WAP *client* yang memiliki fungsi mengirim permintaan, kemudian *gateway* sebagai penterjemah antara WAP dan HTTP serta *server* yang memproses permintaan dari gateway akan memberikan jawaban permintaan tersebut [KOM-03:01-03].

Mobile Information WAP adalah suatu sistem informasi bergerak yang bertujuan untuk memberi kemudahan seseorang yang ingin mendapatkan suatu informasi tanpa dibatasi waktu, ruang serta letak geografis [MAR-03:05].

Proses pemesanan Jasa Travel di Malang masih menggunakan cara manual. Selain dengan datang ke tempat yang bersangkutan, pemesanan Jasa Travel juga dapat dilakukan melalui telepon. Untuk pemesanan Jasa Perjalanan Travel secara langsung, pelanggan memilih tempat tujuan, jam keberangkatan dan membayar ditempat. Begitu juga dengan pemesanan jasa kurir barang, pelanggan datang membawa barang yang akan dikirim, mengisi formulir pengiriman barang dan membayar biaya pengiriman ditempat. Untuk masalah pembayaran dapat dilakukan saat penjemputan oleh pihak travel atau melalui transfer bank ke rekening pihak travel.

Sistem pemesanan travel menggunakan WAP ini merupakan solusi permasalahan yang terjadi pada Biro Perjalanan. Permasalahan yang dirangkum dan akan dapat diselesaikan dengan adanya sistem pemesanan menggunakan WAP ini adalah :

Sistem Pemesanan Manual :

- Pelanggan : memesan jasa perjalanan dan kurir secara manual pada kantor Biro Perjalanan.
- Biro Perjalanan : pendataan transaksi yang terjadi juga harus dilakukan secara manual, shg akan menimbulkan kesulitan dan kurangnya ketelitian jika dibandingkan dengan pencatatan secara otomatis.

Sistem Pemesanan melalui Website :

- Biro Perjalanan : jika pelanggan memesan jasa melalui website, maka Biro Perjalanan tidak perlu melakukan pendataan transaksi karena telah tercatat secara otomatis
- Pelanggan : pemesanan melalui website tidak efisien dan menyita waktu pelanggan dikarenakan harus terhubung dan membuka situs pemesanan melalui internet

Keuntungan bagi pelanggan dengan adanya sistem WAP ini adalah pemesanan secara mobile, kapan saja dan dimana saja asalkan ada jaringan GPRS dan handphone. Untuk pendataan dapat dilakukan secara otomatis seperti halnya pemesanan menggunakan website.

Penggunaan jaringan internet khususnya layanan WAP sebagai alternatif media pemesanan jasa perjalanan akan memberikan keuntungan bagi pengguna maupun

penyedia jasa karena begitu banyak penghematan, baik waktu maupun biaya. Terutama bagi penyedia jasa dilihat dari segi perebutan pangsa pasar.

2.1 **Global System for Mobile Communication (GSM).**

GSM (*Global System for Mobile Communication*) merupakan standar komunikasi seluler digital yang telah diterima secara global [IEC-08]. Berbeda dari sistem nirkabel generasi sebelumnya (1G), GSM menggunakan teknologi digital dan metode transmisi TDMA (*Time Division Multiple Access*), dimana penggunaan kanal tidak diperuntukkan bagi satu *user* saja.

Secara umum, *network elemen* dalam arsitektur jaringan GSM terdiri atas [IEC-08] :

- **Mobile Station (MS)**

Mobile Station adalah perangkat yang digunakan oleh pelanggan untuk melakukan pembicaraan. Secara umum sebuah *mobile system* terdiri dari *Mobile Equipment (handset)* dan SIM (*Subscriber Identity Module*) Card.

- **Base Station Sub-system (BSS)**

Secara umum BSS terdiri dari:

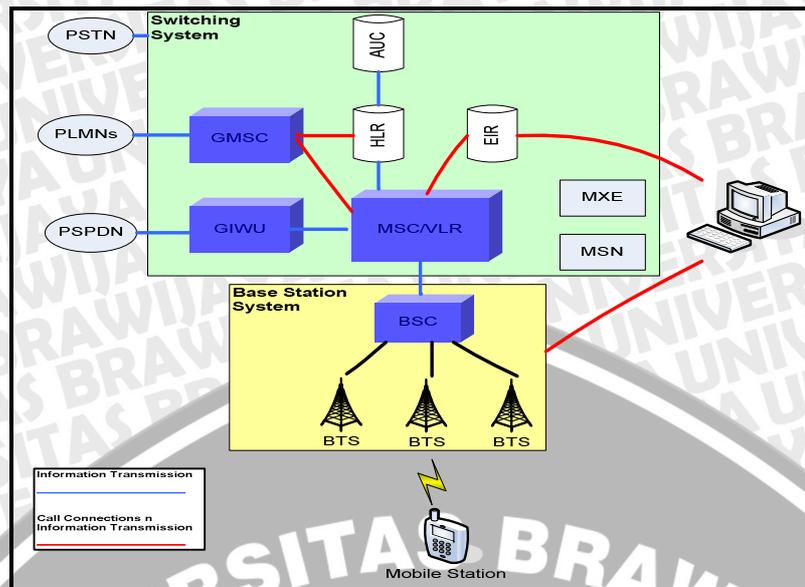
1. *Base Transceiver Station (BTS)* , adalah perangkat GSM yang berhubungan langsung dengan *Mobile Station*, yang berfungsi sebagai pengirim dan penerima (*transceiver*) sinyal komunikasi dari/ke MS, dan menyediakan *radio interface* antara MS dan jaringan GSM. Karena fungsinya sebagai *transceiver*, maka bentuk fisik sebuah BTS adalah *tower* dengan dilengkapi antena sebagai *transceiver*. Sebuah BTS dapat meng-*cover* area sejauh 35 km.
2. *Base Station Controller (BSC)*, adalah perangkat yang mengontrol kerja BTS-BTS yang secara hierarki berada di bawahnya.

- **Network Sub-System (NSS)**

1. *Mobile Switching Center (MSC)*, adalah pusat *network element* dalam sebuah jaringan GSM. Semua hubungan (*voice call / transfer data*) yang dilakukan oleh *mobile subscriber* selalu menggunakan MSC sebagai pusat pembangunan hubungannya.
2. *Home Location Register (HLR)*, adalah *network element* yang berfungsi sebagai sebuah *database* untuk menyimpan semua data dan informasi

mengenai pelanggan yang tersimpan secara permanen, dalam arti tidak tergantung pada posisi pelanggan.

3. *Visitor Location Register (VLR)*, adalah *network element* yang berfungsi sebagai sebuah *database* yang menyimpan data dan informasi pelanggan, dimulai pada saat pelanggan memasuki suatu area yang bernaung dalam wilayah MSC VLR (setiap MSC akan memiliki 1 VLR sendiri) tersebut (melakukan *roaming*). Informasi pelanggan yang ada di VLR ini pada dasarnya adalah salinan dari informasi pelanggan yang ada di HLR-nya. Adanya informasi mengenai pelanggan dalam VLR memungkinkan MSC untuk melakukan hubungan baik *incoming call* (panggilan masuk) maupun *outgoing call* (panggilan keluar). VLR bertindak sebagai *database* pelanggan yang bersifat dinamis, karena selalu berubah setiap waktu, menyesuaikan dengan pelanggan yang memasuki atau berpindah dalam suatu area cakupan suatu MSC. Data yang tersimpan dalam VLR secara otomatis akan selalu berubah mengikuti pergerakan pelanggan. Ketika pelanggan bergerak meninggalkan area suatu MSC dan menuju area MSC lainnya, maka informasinya akan dicatat di VLR MSC baru dan dihapus dari VLR sebelumnya. Dengan demikian posisi pelanggan dapat dimonitor secara terus menerus dan hal ini akan memungkinkan MSC untuk melakukan penyambungan pembicaraan / SMS dari / ke pelanggan ini dengan pelanggan lain.
4. *Authentication Center (AuC)*, menyimpan semua informasi yang diperlukan untuk memeriksa keabsahan pelanggan, sehingga usaha untuk mencoba mengadakan hubungan pembicaraan bagi pelanggan yang tidak sah dapat dihindarkan.



Gambar 2.4 GSM Network

Sumber : [IEC-08 : 04]

- **Operation and Support System (OSS)**

OSS sering juga disebut dengan OMC (*Operation and Maintenance Center*), adalah sub-sistem jaringan GSM yang berfungsi sebagai pusat pengendalian dan *maintenance* perangkat (*network element*) GSM yang terhubung dengannya

2.1.1 Teknologi Tanpa Kabel untuk Teknologi Selular.

Ada beberapa teknologi tanpa kabel untuk teknologi selular ini, diantaranya adalah [STI-09 : 02] :

- **CDMA (Code Division Multiple Access)**, menggunakan teknologi spread-spectrum untuk mengedarkan sinyal informasi yang melalui bandwidth yang lebar (1,25 MHz). Teknologi ini awalnya dibuat untuk kepentingan militer, menggunakan kode digital yang unik, lebih baik daripada channel atau frekuensi RF.
- **AMPS (Advanced Mobil Phone Service)** merupakan teknologi analog yang menggunakan FDMA (*Frequency Division Multiple Access*) untuk membagi-bagi bandwidth radio yang tersedia ke pada sejumlah channel diskrit yang tetap. Dengan AMPS, bandwidth 1,25 MHz yang diberikan untuk penggunaan selular dibagi menjadi channel dengan lebar 30 KHz, masing-masing hanya dapat melayani satu subscriber pada satu waktu. Satu subscriber mengakses sebuah channel maka tidak satupun subscriber lainnya dapat mengakses channel tersebut sampai panggilan pertama itu berhenti atau handed-off ke base station lainnya.

- **TDMA (*Time Division Multiple Data*)**, merupakan sebuah teknologi digital, sama halnya yaitu dengan membagi-bagi spektrum yang tersedia kepada sejumlah channel diskrit yang tetap, meskipun masing-masing channel merepresentasikan time slot yang tetap daripada band frekuensi yang tetap. Sebagai contoh yang mengimplementasikan teknologi TDMA adalah GSM, yang membagi carriers berlebar 2300 KHz menjadi delapan time-division channel. GSM (*global sistem for mobile*) adalah teknologi yang berbasis TDMA.
- **UMTS (*Universal Mobile Telecommunication Access*)** merupakan salah sistem generasi ketiga yang dikembangkan di Eropa. dirancang sehingga dapat menyediakan bandwidth sebesar 2 Mbits/s. Layanan yang dapat diberikan UMTS diupayakan dapat memenuhi permintaan pemakai dimanapun berada, artinya UMTS diharapkan dapat melayani area yang seluas mungkin, jika tidak ada cell UMTS pada suatu daerah dapat di route-kan melalui satelit. Frekuensi radio yang dialokasikan untuk UMTS adalah 1885-2025 MHz dan 2110-2200 MHz. Pita tersebut akan digunakan oleh cell yang kecil (*pico cell*) sehingga dapat memberikan kapasitas yang besar pada UMTS.

2.2 **General Packet Radio Service (GPRS).**

General Packet Radio Service (GPRS) menggunakan prinsip sebuah paket radio untuk mentransfer paket – paket data *user* ke dalam jalur yang efisien antara stasiun – stasiun GSM *Mobile* paket data luar jaringan. Pengubah paket adalah dimana paket data dipecah ke dalam paket – paket yang ditransmisikan secara terpisah kemudian digabungkan kembali pada saat di akhir penerimaan. GPRS memberikan set-up koneksi hampir secara instant dan secara kontinyu menghubungkan dengan internet. Kecepatan data dari GPRS mempunyai range dari 14.4 Kb/s sampai 115 Kb/s [UCT-09 : 02].

Beberapa keuntungan menggunakan jaringan GPRS di pihak *user* adalah :

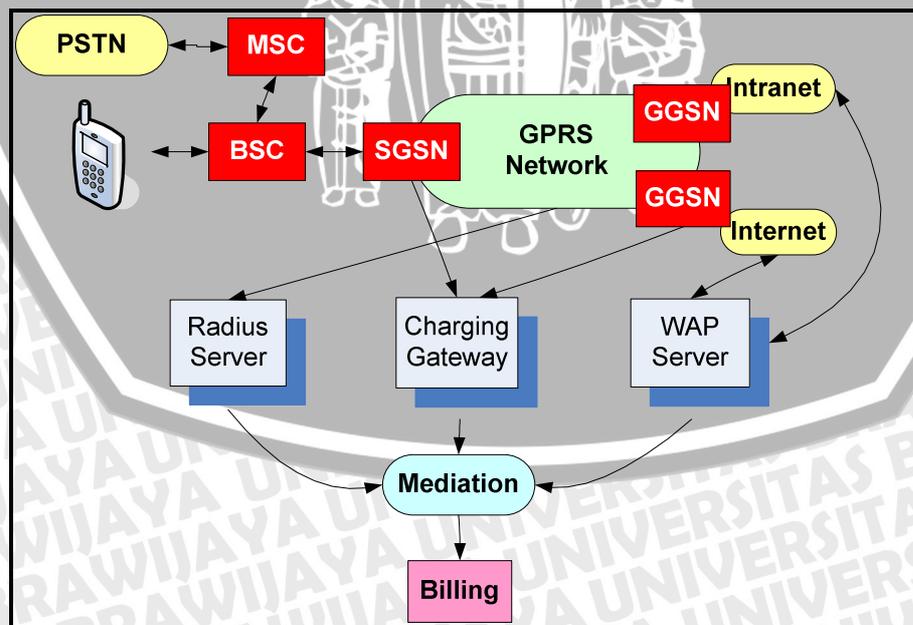
1. *Bandwidth* yang besar, sehingga transmisi data juga cepat.
2. Komunikasi data dan suara secara simultan.
3. Terkoneksi ke internet secara konstan.
4. Hanya mengenakan biaya ketika data ditransmisikan, tidak berdasarkan durasi hubungan.

Secara rinci ada beberapa faktor yang menjadi pertimbangan bahwa GPRS merupakan teknologi kunci untuk data bergerak, yakni [UCT-09 : 02] :

1. Mampu memanfaatkan kemampuan cakupan global yang dimiliki GSM, sebagai generasi kedua (2G) sistem *mobile phone*.
2. Memperkaya utiliti investasi untuk perangkat GSM yang sudah ada
3. Merupakan teknologi jembatan yang bagus menuju generasi ketiga (3G)
4. Berbasis paket data yang lebih efisien dalam penggunaan sumber daya
5. Memiliki laju data sampai 115 kbps yang berarti sekitar dua kali lipat daripada koneksi 'dial up' 56 kbps pada jaringan komunikasi saat ini.

Beberapa layanan komunikasi data yang berkembang dengan adanya jaringan GPRS ini antara lain :

1. MMS (*Multimedia Messaging System*), dengan MMS ini *user* bisa mengirimkan pesan dalam bentuk multimedia (suara, klip video, gambar)
2. *Traffic Monitoring*, dengan layanan ini *user* bisa melihat keadaan lalu lintas di suatu tempat secara *real time* dengan menggunakan kamera *traffic monitoring*.
3. VOIP (*Voice Over IP*), layanan ini biasanya digunakan antar *user* PDA (*Personal Digital Assistant*). Pemakai PDA pertama harus menginstal suatu program terlebih dahulu baru bisa menggunakan VOIP.



Gambar 2.5 Arsitektur GPRS

Sumber : [UCT-09 : 17]

Keterangan :

- **Mobile Station (MS)**
MS dapat dikatakan perangkat selular yang terhubung langsung dengan jaringan GSM, yaitu *SIM (Subscriber Identify Module) Card* dan perangkat keras seperti telepon selular, PDA, perangkat komputer yang terhubung menggunakan jaringan GPRS.
- **Base Station System (BSS)**
BSS terdiri dari BTS (*Base Transceiver Station*) dan BSC (*Base Station Controller*). BSC menyediakan semua fungsi control dan relasi fisik antara MSC (*Mobile Switching Centre*) dan BTS. BTS menangani *interface* radio pada *mobile station*. Sekumpulan BTS dikendalikan oleh sebuah BSC.
- **Gateway Mobile Switching Centre (GMSC)**
GMSC menyediakan fungsi pembatas diantara PLMN (*Public Line Mobile Network*). GMSC memetakan format *traffic* dan pensinyalan PSTN (*Public Switched Telephone Network*) dan mengubahnya menjadi protokol pada jaringan *mobile*.
- **Public Switched Telephone Network (PSTN)**
PSTN merupakan istilah umum untuk layanan dan jaringan telepon.
- **Home Location Register (HLR)**
HLR adalah basis data yang menyimpan data *user* jaringan GPRS.
- **Serving GPRS Support Node (SGSN)**
SGSN adalah komponen utama jaringan GPRS. SGSN akan meneruskan paket data dari/ke MS.
- **Gateway GPRS Support (GGSN)**
GGSN juga merupakan komponen utama jaringan GPRS. GGSN mengubah paket data GSM dari SGSN menjadi paket TCP/IP.
- **Authentication Center (AuC)**
AuC adalah basis data yang berisi informasi *user* yang diperbolehkan memakai jaringan GPRS. AuC merupakan bagian dari HLR.
- **GPRS backbone networks**
GPRS *backbone network* adalah Intranet dari jaringan GPRS.
- **Border Gateway (BG)**

BG adalah *server* yang menjadi perantara antara dua operator.

2.2.1 Wireless Application Protocol (WAP).

Wireless Application Protocol atau WAP adalah standar internasional untuk aplikasi yang menggunakan komunikasi nirkabel (*wireless*). Aplikasi utamanya adalah agar Internet dapat diakses melalui *mobile phone* atau PDA [TEL-04 : 02].

WAP *browser* didesain untuk menyediakan semua pelayanan dasar *web browser* berbasis komputer namun telah disederhanakan untuk beroperasi didalam keterbatasan pada *mobile phone*. Saat ini WAP merupakan protokol yang digunakan mayoritas situs Internet *mobile* diseluruh dunia, yang dikenal sebagai situs WAP (*WAP sites*) [TEL-04 : 02-03].

Komunikasi Internet lewat WAP menggunakan sebuah tahapan yang disebut sebagai *WAP Gateway*. Tahapan ini berada diantara ponsel dan server. Alur komunikasi internet lewat WAP yang terjadi secara lengkap adalah saat ponsel ingin meminta sebuah informasi yang ada di server, ia harus melewati dahulu *WAP Gateway*.

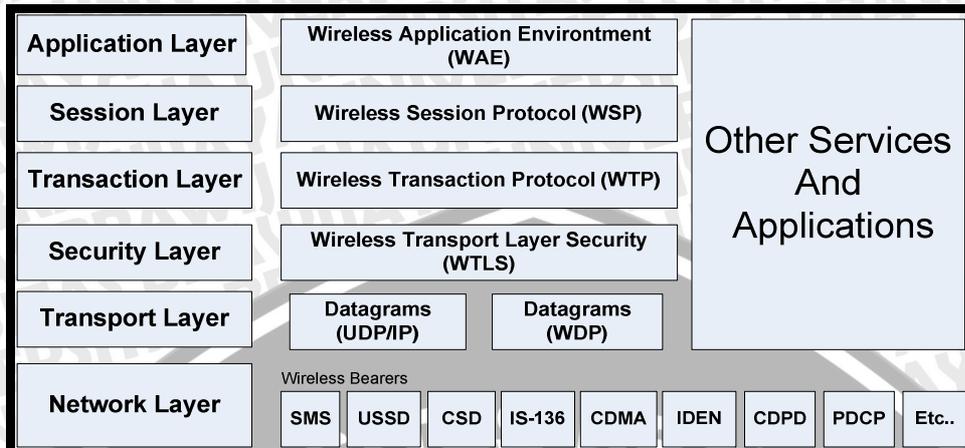
Fungsi *WAP Gateway* adalah untuk meneruskan permintaan informasi dari ponsel menuju server lewat *HTTP Request*, dan sebaliknya, dari server menuju ke ponsel lewat *HTTP Response* [AGU-04 : 05-06].

WAP Gateway berfungsi sebagai penerjemah informasi dari *content server* untuk ditampilkan pada perangkat *mobile client* serta sebaliknya. *WAP Gateway* juga dapat berfungsi sebagai proxy. Fungsi *WAP Gateway* / proxy antara lain dapat dijabarkan sebagai berikut [AHM-09 : 67] :

- Sebagai *interface* penghubung protokol WAP dengan protokol internet.
- Sebagai *caches* untuk *header* protokol yang memperkuat efisiensi transfer data.
- *Caching content* untuk *file overload* dari *application server*.
- Domain Name Server (DNS) dari client yang akan memetakan URL ke *IP address* tujuan.
- Sebagai *security gateway* di mana dilakukan autentikasi client / pengguna sebagai *subscriber* suatu layanan WAP.
- Sebagai *billing support* yang menjadi tempat informasi mengenai *client* yang melakukan pembayaran melalui *mobile-commerce*.

WAP Server mempunyai fungsi untuk menyediakan *file* bertipe WML dan WML *script*. Di *server* ini juga dapat dijalankan program *servlets* yang akan menambah kemampuan aplikasi sebagai *dynamic WAP-content*. Bahasa pemrograman yang dapat

dipakai seperti Java, ASP, Perl, CGI, dan lain-lain. *Layer-layer* protokol WAP sendiri terdiri atas :



Gambar 2.6 Layer Protokol WAP

Sumber : [AHM-09 : 21]

- **Wireless Application Environment (WAE)**
WAE terdiri atas WML yang merupakan *content* informasi yang akan ditampilkan oleh *microbrowser* pada layar telepon seluler [AHM-09 : 21-23]
- **Wireless Session Protocol (WSP)**
WSP disediakan sebagai cara mengatur penukaran *content* / data antara *client* dengan *server*
- **Wireless Transaction Protocol (WTP)**
WTP memastikan paket data terkirim sesuai yang diminta
- **Wireless Transport Layer Security (WTLS)**
WTLS mengkompresi dan mengenkripsi data yang dikirim supaya aman
- **Wireless Datagram Protocol (WDP)**
WDP mengirim data dan mengkoreksi kesalahan pengiriman data
- **Wireless Bearers**
Merupakan jaringan *wireless* pembawa (*carrier*) WAP.

2.2.2 Wireless Markup Language (WML).

Wireless Mark-up Language merupakan bahasa mark-up yang berbasis pada *Extensible Markup Language* (XML). Sebuah halaman WML terdiri dari bagian *header* dan *body*. Dalam *header* terdapat dua hal yang harus dideklarasikan, yaitu versi XML yang akan digunakan dan dideklarasikan *Document Type Definition* (DTD) [AGU-04 :

14]. Pendeklarasian DTD ini bertujuan agar tipe data dalam dokumen yang dibuat dapat dikenali dengan benar. Deklarasi WML adalah sebagai berikut [AGU-04 : 14] :

```
<?xml version="1.0"?><!DOCTYPE wml PUBLIC "-//WAPFORUM//DTDWML1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

Semua *tag* WML memiliki atribut sebagai standar : *id*, yang digunakan sebagai referensi unik dalam sebuah *deck*, dan *class*, yang digunakan pada sisi server. Bagian *body file* WML diawali dan diakhiri oleh pasangan *tag* `<wml>` dan `</wml>`. Pasangan *tag* ini disebut dengan *deck*. Dalam satu *deck* dapat terdiri dari satu *template* dan satu atau lebih *card*. *Tag template* digunakan sebagai cetak biru *event* semua *card* dalam satu *deck*. Sebuah *card* dapat berisi satu atau lebih komponen sebagai berikut [AGU-04 : 14-18] :

1. Teks terformat, dapat berupa teks, gambar, dan link.
2. Elemen `<input>`, untuk menerima input dari user berupa string.
3. Elemen `<select>`, user dapat memilih dari daftar pilihan.
4. Elemen `<fieldset>`, bertindak sebagai container bagi elemen lain

Level sintaks yang menyusun suatu *deck* adalah sebagai berikut :

```
<wml>
    <!--keterangan kode dalam tag ini -->
    <head>
        kontrol akses data dan informasi meta
    </head>
    <template>
        event pada level deck
    </template>
    <card>
        event pada level card
    </card>
</wml>
```

Deskripsi dari tag – tag dasar yang menyusun *deck* WML dapat dilihat pada tabel di bawah ini :

Tabel 1 *Tag – tag* dasar *deck* WML

Elemen	Keterangan
--------	------------



<wml>	Elemen harus ada. Elemen ini menentukan sebuah deck.
<head>	Elemen untuk menyatakan informasi keseluruhan dari sebuah deck, termasuk metadata dan kontrol akses.
<card>	Satu atau lebih <card> menentukan dan action user interface untuk deck.

Sumber : [AGU-04 : 14-18]

2.2.2.1 Elemen *Deck* dan *Card*.

WML didesain untuk *bandwidth* kecil, dan perangkat berpenampilan kecil sehingga dipakai konsep setumpuk kartu. Sebuah dokumen WML dianggap sebagai tumpukan kartu, sehingga memungkinkan lebih dari satu kartu bisa di-*download* oleh klien dalam sekali pengambilan data.

Elemen-elemen dalam *Deck* dan *Card* [AGU-04 : 14-25] :

1. Elemen WML

Elemen ini mewakili permulaan dan akhir dari suatu halaman dan mendefinisikan tumpukan kartu, membuka dan menutup semua informasi di dalam *deck*. Cara penulisan elemen ini, dibuka dengan sintaks <wml> dan ditutup dengan menuliskan </wml>.

2. Elemen *Template*

Elemen *template* ini mendeklarasikan ketetapan untuk semua *card* di dalam *deck*. Cara penulisan sintaksnya dibuka dengan <template> dan ditutup oleh </template>.

3. Elemen *Card*

Mengindikasikan tampilan yang keluar pada layar klien. Elemen ini mewakili permulaan dan akhir dari *card*. Cara penulisan elemen ini, dibuka dengan <card> dan ditutup dengan </card>.

4. Elemen *Head*

Elemen *head* memberikan informasi yang berkaitan kedalam *deck*. Penulisannya dibuka dengan <head> dan ditutup dengan </head>. Didalam elemen *head* terdapat elemen *meta* yang berfungsi untuk pengklarifikasian suatu *deck* ke dalam *search engine*.

2.2.2.2 Element *Event*.

Beberapa elemen WML membuat suatu aksi dengan menggunakan elemen-elemen *event*.

Elemen – elemen *event* adalah :

1. Elemen *do*

Sintaks ini memberikan pilihan pada *user* untuk melakukan *action*. Cara penulisan elemen ini dibuka dengan sintaks `<do>` dan diakhiri dengan sintaks `</do>`.

2. Elemen *Timer*

Elemen *timer* mendeklarasikan waktu pada kartu, dimana waktunya mulai ketika memasuki *card*, dan berhenti setelah waktu yang ditentukan berakhir atau ketika keluar dari *card* dimana *timer* tersebut aktif.

3. Elemen *Onpick*

Dalam pilihan lebih dari satu, dibutuhkan sintaks *onpick*. Sehingga pengguna dapat memilih atau tidak memilih salah satu dari pilihan-pilihan tersebut. Penulisiannya yaitu:

```
<option onpick="alamat">pilihan satu</option>
```

2.2.2.3 Elemen *Task*.

Elemen *do* dapat menspesifikasikan *task* yang akan dijalankan pada klien, ketika pengguna bernavigasi pada klien menuju *deck* atau *card* yang lain.

Elemen – elemen *task* adalah :

1. Elemen *Go*

Elemen ini akan menavigasikan klien ke lokasi yang telah ditentukan oleh *href*. Cara penulisannya adalah `<go href="alamatnya"/>`

2. Elemen *Prev*

Elemen ini menavigasikan kembali ke halaman sebelumnya. Elemen *prev* mengambil alamat sebelumnya dari fasilitas *history*. Cara penulisannya adalah `<prev/>`

3. Elemen *Refresh*

Elemen ini berfungsi untuk membuka kembali *card* yang terbuka, serta meng-*update* konteks yang muncul jika terjadi perubahan. Cara penulisan elemen ini yaitu `<refresh/>`

4. Elemen *Noop*

Elemen *noop* berarti *no operation*, elemen tidak mengerjakan apa-apa. Fungsi elemen ini jika ingin mengesampingkan efek elemen *template*. Cara penulisan elemen ini yaitu `<noop/>`

2.2.2.4 Elemen Variabel.

User dapat membuat *card* dinamis yang fleksibel, mengganti isi tampilan, dan menavigasi berdasarkan input. Variabel WML dapat digunakan sebagai pengganti *string-string*.

2.2.2.5 Elemen User Input.

Elemen – elemen ini membantu mekanisme pengambilan input dari *user*.

Elemen – elemen ini adalah :

1. Elemen *Input*

Elemen ini menspesifikasi masukan. Sehingga dapat menentukan format masukan yang diisi oleh *user* dan dapat mengisi *text* standar pada *field*.

2. Elemen *Select*

Elemen *select* diperlukan agar *user* dapat memilih pilihan – pilihan yang ada, baik pilihan tunggal ataupun pilihan lebih dari satu. Cara penulisan elemen ini dibuka dengan sintaks `<select>` dan ditutup dengan sintaks `</select>`.

3. Elemen *Option*

Elemen ini berada didalam elemen *select*, yang menspesifikasikan pilihan tunggal yang dipilih *user*. Cara penulisan elemen ini dibuka dengan sintaks `<option>` dan ditutup dengan sintaks `</option>`.

2.2.2.6 Elemen *Anchor*.

Elemen *anchor* digunakan untuk menspesifikasikan *link*. Elemen ini mengarahkan *user* kepada *link* yang telah ditentukan. Cara penulisan elemen ini dibuka dengan sintaks `<anchor>` dan ditutup dengan sintaks `</anchor>`.

2.2.2.7 Elemen *Image*.

Elemen *image* berguna untuk menampilkan gambar kedalam alur teks pada klien *user*. Gambar pada WML sudah ditentukan, yaitu berformat "wbmp".

Cara penulisan sintaksnya dibuka dengan `` dan ditutup ``. Elemen – elemen yang terdapat pada elemen *image* adalah : *alt*, *src*, *width* dan *height* serta *align*.

2.2.2.8 Elemen *Format Text*.

Elemen yang terdapat pada elemen ini adalah :

1. *Paragraf*

Elemen ini digunakan untuk memulai paragraf. Penulisannya dibuka dengan

`<p>` dan ditutup dengan `</p>`.

2. *Emphasis*

Emphasis diperlukan untuk memberi penekanan pada teks. Sintaks penekanan adalah sebagai berikut :

- *i*

Menampilkan jenis huruf miring.

- *b*

Menampilkan jenis huruf tebal.

- *u*

Menampilkan jenis huruf yang bergaris bawah.

- *big*

Menampilkan jenis huruf yang lebih besar.

- *small*

Menampilkan jenis huruf yang lebih kecil.

3. Karakter khusus

Karakter khusus dibutuhkan dalam beberapa penulisan WML. Berikut cara penulisannya :

- Kutip dua (")

Bisa ditulis dengan `"`; atau `"`;

- *Ampersand* (&)

Bisa ditulis dengan `&`; atau `&`;

- Kutip satu (')

Bisa ditulis dengan `'`; atau `'`;

- Lebih kurang dari (<)

Bisa ditulis dengan `<`; atau `<`;

- Lebih besar dari (>)

Bisa ditulis dengan `>`; atau `>`;

- Spasi

Bisa ditulis dengan ` `; atau ` `;

2.3 PHP.

PHP merupakan suatu bahasa scripting yang menyatu dengan HTML (aplikasi yang berjalan diatas web browser) yang fungsinya sebagai penghubung antara halaman Web dengan database yang ada di server. PHP ini letaknya berada di *server (server-side HTML-embedded scripting)* artinya *sintak* dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di *server* tetapi bisa disertakan pada halaman HTML biasa. Tujuan dari bahasa scripting ini adalah untuk membuat aplikasi-aplikasi yang dijalankan diatas teknologi Web. Dalam hal ini aplikasi pada umumnya akan memberikan hasil pada *Web Browser* (pengakses *Web*), tetapi prosesnya secara keseluruhan dijalankan di *Web server*. [SUT-07 : 95]

Seluruh aplikasi *Web* dapat dibuat dengan PHP. Namun kekuatan yang paling utama adalah konektifitas database dengan halaman *Web* yang mudah. Kode program PHP menyatu dengan tag-tag HTML dalam satu file. Kode PHP diawali dengan tag `<?>` atau `<?php` dan ditutup dengan `?>`. File yang berisi tag HTML dan kode PHP ini diberi ekstensi `.php` atau ekstensi lainnya yang ditetapkan pada *Apache/web server*. Berdasarkan ekstensi ini, pada saat file diakses, *server* akan tahu bahwa file ini mengandung kode PHP. *Server* akan menterjemahkan kode ini dan menghasilkan output dalam bentuk tag HTML yang akan dikirim ke *browser client* yang akan mengakses file tersebut. Misal :

```
<?
$nama="candra";
//jika memanggil maka harus didahului
dengan perintah echo
echo ("Nama Anda : $nama");
?>
hasilnya ialah :
Nama Anda : candra
```

Adapun kelebihan-kelebihan dari PHP adalah [SUT-07 : 95] :

1. Mempermudah aplikasi web melakukan koneksi dengan database.
2. Gratis dan open source.
3. Sederhana, stabil dan terus dioptimasi.
4. Bersifat multi platform.
5. Keamanan yang baik dan terus diperbaiki.
6. Dengan PHP proses pengembangan aplikasi menjadi mudah karena :

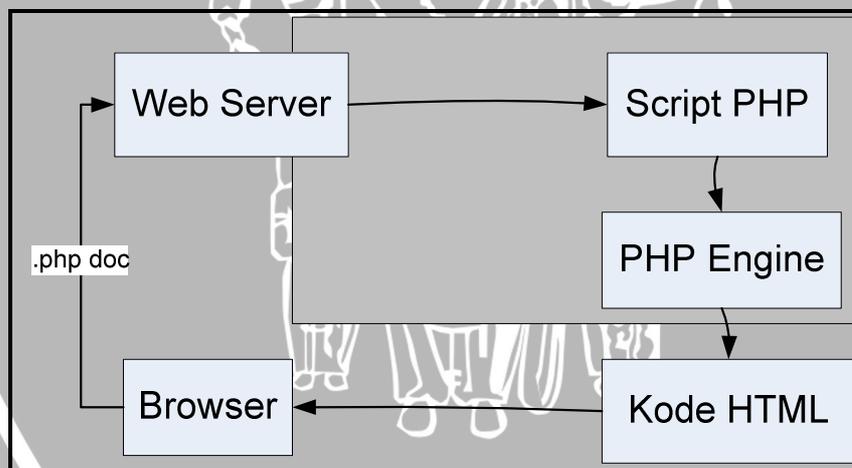
- a. *Script* (kode program) terintegrasi dengan file HTML, sehingga *developer* bisa berkonsentrasi langsung pada penampilan dokumen webnya.
- b. Tidak ada proses *compiling* dan *linking*.
- c. Berorientasi objek (*object oriented*)
- d. Sintaks programnya mudah dipelajari, sangat menyerupai C dan Perl.
- e. Integrasi yang sangat luas ke berbagai server database. Menulis web yang terhubung ke database menjadi sangat sederhana. Database yang didukung oleh PHP adalah : Oracle, Sybase, MySQL, Solid, ODBC, PostgreSQL, Adabas D, FilePro, Velocis, Informix, dBase, Unix dbm.

Untuk membuat program WAP yang berbasis PHP, maka setiap halaman di WAP harus ditambahkan dengan *header* pengenalan sebagai berikut [NUG-05 : 51] :

```
header('Content-type: text/vnd.wap.wml');
```

Selain itu perlu penambahan pengenalan *script* ke dalam versi penulisan PHP, yaitu dengan memberikan parameter *echo* sebagai berikut :

```
echo('<?xml version="1.0"?>');
echo('<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML1.1/EN">');
```



Gambar 2.7 Skema Dasar PHP

Sumber : [ANI-09 : 01]

2.4 Apache Web Server.

Web server adalah merupakan suatu server internet yang menggunakan protocol HTTP untuk melayani semua proses pentransferan data. *Web server* hingga saat ini merupakan server yang dapat dikatakan sebagai tulang punggung bagi semua pengguna



internet. Hal ini dikarenakan, *Web Server* bukan hanya bisa dapat melayani jenis data dalam bentuk text, akan tetapi juga file multimedia dan juga dapat berinteraksi dengan dunia wireless internet dengan menjadikannya sebagai sebuah *Wireless Acces Protocol (WAP) gateway* dan sebagainya. [SYA-04 : 01].

Apache Web Server adalah aplikasi yang paling banyak digunakan untuk sebuah Web server, disebabkan karena beberapa alasan yaitu [SYA-04 : 01] :

- Kecepatan yang dimiliki lebih baik jika dibandingkan dengan palikasi lain yg digunakan untuk web server.
- Performance sangat baik
- Dapat diperoleh dengan gratis

Apache web server memiliki program pendukung cukup banyak yang dapat memberikan layanan yang cukup bagi penggunanya. Berikut ini adalah beberapa program pendukung dari Apache Web Server yaitu [SYA-04 : 02] :

- Kontrol Akses
- *Common Gateway Interface (CGI)*, yang paling terkenal dan sangat sering digunakan adalah Perl (*Practical Extraction and Report Language*).
- PHP (*Personal Home Page*)
- SSI (*Server Side Include*)

2.5 MySQL.

MySQL merupakan Relational Database Management System (DBMS) yang didistribusikan secara gratis, dimana setiap orang bebas untuk menggunakan. Sebagai *database server*, MySQL dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam query data [SUJ-05].

Sebagai *database server* yang memiliki konsep *database* modern MySQL memiliki banyak sekali keistimewaan antara lain [SUJ-05] :

1. Portability, dimana dapat berjalan stabil pada berbagai sistem operasi.
2. Open Source.
3. Performance running, yaitu memiliki kecepatan yang handal.
4. Column Type, Command Function, Security.
5. Scalability yaitu mampu menangani database dalam skala besar.

Perintah-perintah SQL dikelompokkan menjadi 5 macam yaitu [SUJ-05] :

1. Data Definition Language (DDL) adalah perintah SQL yang digunakan untuk menjelaskan objek dari database. Dengan kata lain DDL digunakan untuk mendefinisikan kerangka database. Perintahnya adalah :

- create : untuk membuat/menciptakan objek database.
- alter : untuk memodifikasi/mengubah objek database.
- drop : untuk menghapus objek database.

Objek database yang dimaksud terdiri dari database, tabel, index dan view.

2. Data Manipulation Language (DML) adalah perintah yang digunakan untuk mengoperasikan atau memanipulasi isi database. SQL menyediakan 4 perintah DML yaitu :

- select : digunakan untuk mengambil data dari database.
- delete : digunakan untuk menghapus data dari database
- insert : untuk menambahkan data pada database.
- update : untuk memodifikasi data pada database.

3. Security adalah perintah-perintah yang digunakan untuk menjamin keamanan data. Antara lain terdiri atas :

- grant : memberi akses kepada user tertentu untuk akses ke database.
- revoke : mencabut hak akses dari user.

4. Integrity adalah perintah-perintah yang digunakan untuk menjaga kesatuan data.

Contoh : recover table, untuk memperbaiki tabel pada database.

5. Auxilary adalah perintah-perintah pelengkap atau tambahan seperti : unload dan rename.

BAB III

METODE PENELITIAN

Pada bab ini akan dibahas berbagai tahapan penelitian yang digunakan untuk menyelesaikan penyusunan tugas akhir. Langkah – langkah yang perlu dilakukan untuk dapat merealisasikan penyusunan tugas akhir yang akan dibuat adalah sebagai berikut :

- Studi literatur
- Analisis dan Perancangan
- Implementasi
- Pengujian dan analisis
- Pengambilan kesimpulan
- Penulisan laporan

3.1 Studi Literatur

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

1. Sistem Informasi
2. *Global System for Mobile Communication (GSM)*
3. *General Packet Radio Service (GPRS)*
4. *Wireless Application Protocol (WAP)*
5. *Wireless Mark-up Language (WML)*
6. PHP
7. MySQL
8. *Apache Web Server*
9. Basis Data
10. *Data Flow Diagram (DFD)*

3.2 Analisis dan Perancangan

Perancangan akan dimulai dengan melakukan analisis proses bisnis, pembuatan Standar Operasi Prosedur (SOP), serta analisis kebutuhan yang akan menjembatani kebutuhan perangkat lunak dengan perancangan perangkat lunak. Analisis kebutuhan menggunakan metode analisis terstruktur dengan menggunakan pemodelan DFD.

Perancangan akan dilakukan dengan melakukan desain arsitektur untuk mendapatkan arsitektur dari sistem yang akan dibangun. Pada proses perancangan juga akan dilakukan perancangan dari kebutuhan-kebutuhan yang telah didefinisikan sebelumnya, pembuatan *site-map* dan perancangan *user-interface*.

3.3 Implementasi

Implementasi aplikasi dilakukan dengan mengacu kepada perancangan aplikasi. Implementasi perangkat lunak dilakukan dengan menggunakan bahasa pemrograman berorientasi objek yaitu menggunakan bahasa pemrograman PHP dan basis data MySQL. Untuk pemrograman WAP (*Wireless Application Protocol*) menggunakan bahasa pemrograman WML (*Wireless Markup Language*).

3.4 Pengujian

Pengujian aplikasi yaitu untuk mengetahui kesesuaian analisa kebutuhan yang dibuat dengan implementasi aplikasi. Analisis sistem dilakukan dengan membandingkan sistem aplikasi yang dengan teori yang ada sehingga didapatkan suatu kesimpulan.

Pengujian dilakukan untuk menemukan kesalahan aplikasi yang dibuat, sehingga sebelum aplikasi yang dihasilkan digunakan oleh pengguna diharapkan didapatkan sesedikit mungkin atau bahkan tidak ditemukan kesalahan.

Proses pengujian dilakukan melalui tiga tahapan (strategi) yaitu pengujian unit, pengujian integrasi dan pengujian validasi. Pada pengujian unit dan pengujian integrasi, akan digunakan teknik pengujian *White Box*. Pada pengujian validasi akan digunakan teknik pengujian *Black Box*. Proses analisis dilakukan dengan membandingkan sistem aplikasi yang dengan teori yang ada sehingga didapatkan suatu kesimpulan.

3.5 Pengambilan Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi dan pengujian sistem aplikasi telah selesai dilakukan dan didasarkan pada kesesuaian antara teori dan praktek. Kesimpulan diambil untuk menjawab rumusan masalah yang telah ditetapkan sebelumnya. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan aplikasi selanjutnya.

3.6 Penulisan laporan

Merupakan kegiatan penulisan laporan terhadap Tugas Akhir yang dibuat yang diharapkan dapat dimanfaatkan bagi pihak-pihak lain dan tidak menutup kemungkinan dapat dilakukan perbaikan dari saran-saran yang diajukan.

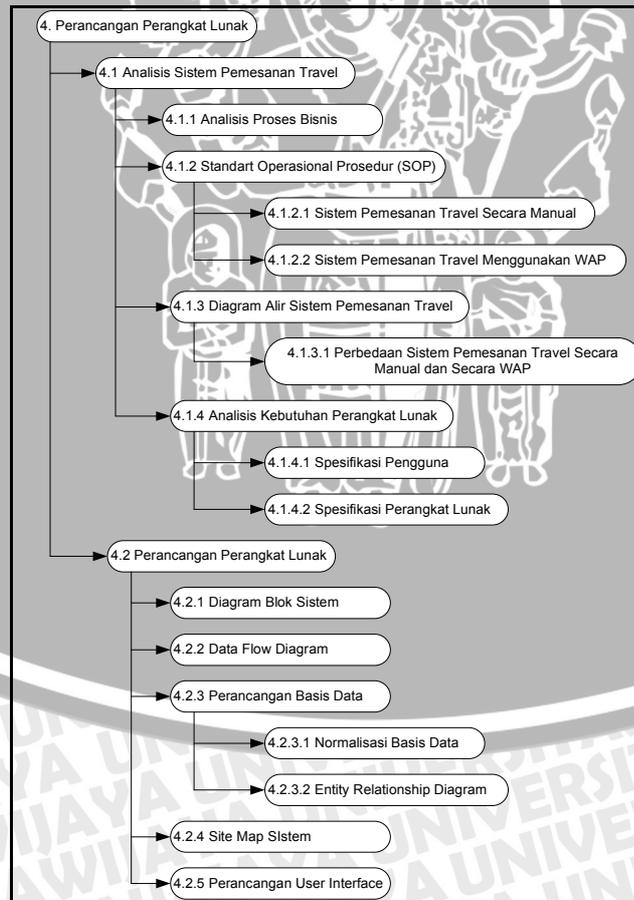


BAB IV PERANCANGAN PERANGKAT LUNAK

Bab ini menjelaskan analisis dan perancangan aplikasi Sistem Pemesanan Jasa Travel dan Pengiriman Barang Menggunakan Travel dengan Layanan WAP. Analisis dibutuhkan untuk menentukan *user* dan memaparkan kebutuhan *user* dalam pemakaian perangkat lunak yang akan dibuat.

Proses analisis dalam pembuatan sistem ini meliputi penjelasan cara kerja sistem yang akan dirancang dalam bentuk analisis proses bisnis, Standar Operasional Prosedur (SOP), dan analisis kebutuhan perangkat lunak.

Tahap perancangan perangkat lunak terdiri dari empat bagian, yaitu implementasi analisis kebutuhan perangkat lunak dalam bentuk *data flow diagram* (DFD), perancangan basis data, *site map* sistem, dan perancangan *Antarmuka User*. Diagram pohon perancangan ditunjukkan dalam Gambar 4.1



Gambar 4.1 Diagram Pohon Perancangan Sistem

Sumber : Analisis dan Perancangan



4.1.1 Analisis Proses Bisnis

Analisis proses bisnis dibutuhkan untuk memahami alur kerja dari proses pemesanan Jasa Travel melalui telepon genggam berbasis WAP ini, agar proses-proses berikutnya pada bagian perancangan ini akan sesuai dengan alur kerja yang telah dibuat sebelumnya.

Pada program yang akan dibuat ini, sistem akan berjalan kurang lebih sebagai berikut:

1. Calon penumpang (anggota) dapat melakukan pemesanan jasa travel melalui telepon genggam yang mempunyai fasilitas layanan WAP dan terdapat jaringan GPRS.
2. Untuk calon penumpang (pengunjung) yang baru pertama kali menggunakan fasilitas pemesanan melalui telepon genggam ini, wajib melakukan registrasi untuk menjadi anggota.
3. Setelah mendaftarkan diri menjadi anggota, akan didapatkan *nick user* dan *password* yang digunakan untuk *login* dalam melakukan pemesanan jasa, pembatalan jasa, atau melihat informasi transaksi *user* bersangkutan.
4. Untuk setiap kota tujuan yang dipesan, maka anggota akan mendapatkan kode transaksi yang berbeda.
5. Anggota diijinkan untuk melakukan pembatalan atas jasa yang telah dipesan dengan memilih transaksi yang akan dibatalkan berdasarkan kode transaksi.
6. Anggota dapat melihat kembali informasi jasa travel yang telah dipesan dengan memilih kode transaksi jasa travel yang masih berlaku setelah sebelumnya *login* terlebih dahulu.
7. Selain itu, pengunjung dapat melihat informasi seputar jasa travel (harga pemesanan per kota tujuan, jenis kendaraan, dan lain-lain) tanpa harus *login* terlebih dahulu.
8. Sistem Pembayaran dilakukan dengan cara COD (*Cash On Delivery*) yaitu system pembayaran saat penjemputan pelanggan atau system pembayaran saat pelanggan sudah sampai ke tempat tujuan.

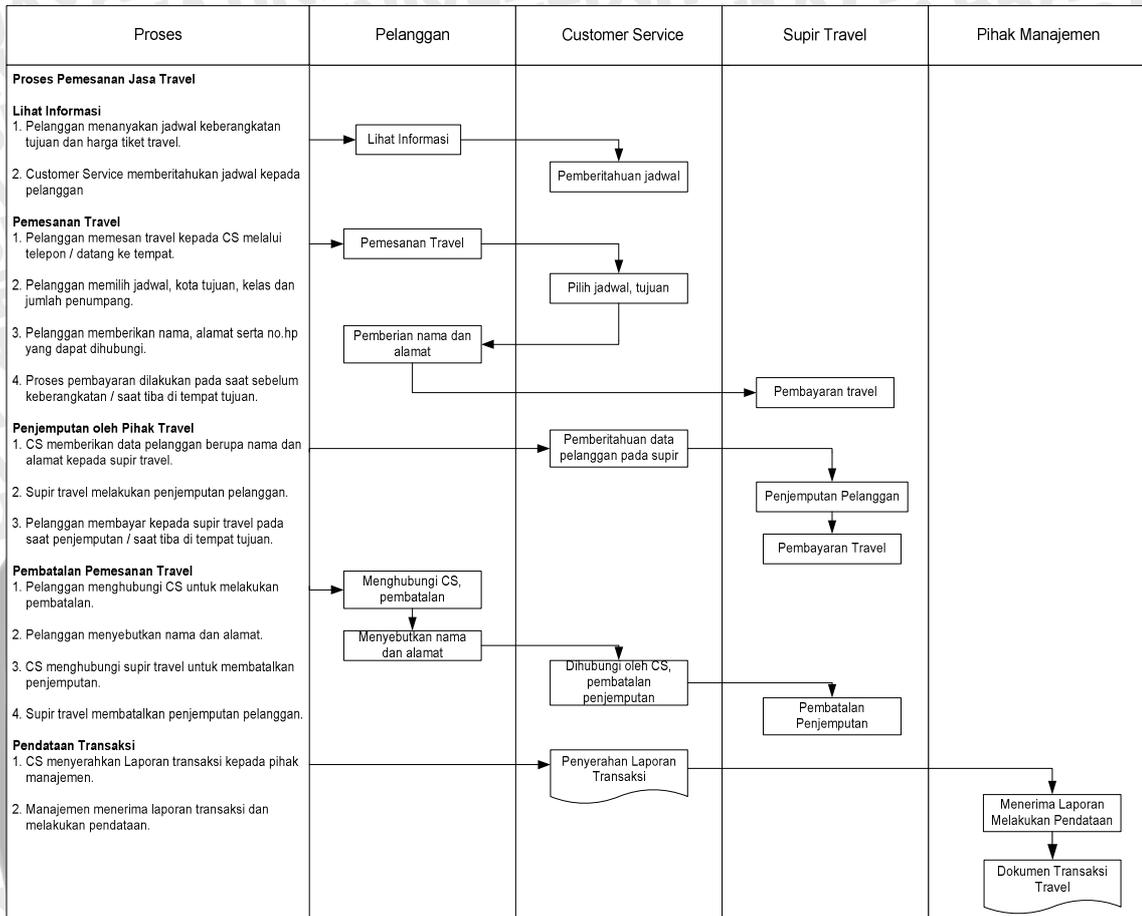
4.1.2 Standart Operational Procedure (SOP)

Standart Operational Procedure (SOP) adalah suatu prosedur standar yang harus dilakukan dalam kegiatan pemesanan jasa travel. SOP sistem ini terdiri dari SOP pemesanan secara manual dan SOP pemesanan menggunakan WAP.

4.1.2.1 Sistem Pemesanan Jasa Travel Secara Manual

SOP manual pemesanan jasa travel ditunjukkan pada tabel dibawah ini :

Tabel 4.1 Standar Operasional Prosedur Pemesanan Jasa Travel Manual



Sumber : Jasa Travel Tirta Jaya Malang

Berikut dipaparkan SOP pemesanan Jasa Travel secara manual :

a. Pelanggan

- Pelanggan tidak disyaratkan mempunyai kualifikasi khusus.
- Pelanggan merupakan individu, bisa siapa saja yang menginginkan informasi pemesanan jasa travel, atau melakukan pemesanan jasa travel.

b. Mekanisme Pemesanan Jasa

- Pelanggan melakukan pemesanan jasa melalui telepon atau datang ke kantor jasa travel yang bersangkutan.
- Pelanggan memilih tanggal keberangkatan, kota tujuan, kelas kendaraan dan jumlah penumpang. Selanjutnya membayar jasa travel tersebut sesuai dengan harga yang telah ditetapkan.

- Pihak travel akan melakukan penjemputan kepada pelanggan sesuai alamat yang telah disebutkan.
- Pembayaran dilakukan melalui supir travel saat penjemputan atau saat tiba di tempat tujuan.
- Pelanggan dipersilahkan menaiki kendaraan travel, dan berangkat menuju kota tujuan yang telah dipesan sebelumnya.

c. Mekanisme Pembatalan Pemesanan

- Pelanggan melakukan pembatalan jasa dengan cara menghubungi *Customer Service* pihak travel yang bersangkutan.
- Pelanggan memberitahukan nama dan alamat yang akan dibatalkan.
- *Customer Service* pihak travel akan menghubungi supir travel untuk membatalkan penjemputan kepada pelanggan.

d. *Customer Service*

- CS tidak disyaratkan mempunyai kualifikasi khusus.
- CS merupakan individu, bisa siapa saja yang berusia di atas 18 tahun dan bersedia bekerja sungguh-sungguh.
- CS mempunyai tugas melakukan transaksi dengan pelanggan dan menyerahkan laporan – laporan kepada manajemen (pengelola perusahaan).

e. Supir Travel

- Supir Travel harus mempunyai SIM A.
- Supir Travel merupakan individu, bisa siapa saja yang berusia di atas 18 tahun dan bersedia bekerja sungguh-sungguh.
- Supir Travel harus sehat, mengetahui jalan-jalan dari kota tempat tujuan dan sabar.
- Supir Travel mempunyai tugas untuk menjemput pelanggan yang telah memesan dan mengantarkannya ke tempat tujuan.

e. Manajemen (Pengelola Perusahaan)

- Manajemen menerima laporan-laporan yang diberikan oleh CS.
- Manajemen berhak untuk melarang atau mencegah terhadap data-data yang akan diberikan kepada pelanggan.
- Manajemen mengontrol perkembangan perusahaan berdasarkan laporan – laporan yang diterima.

Berikut dipaparkan SOP pemesanan Jasa Travel menggunakan layanan WAP :

a. User Pengunjung

- User pengunjung merupakan individu yang ingin mendapatkan informasi pemesanan jasa travel, yaitu meliputi informasi harga pemesanan jasa, persyaratan keanggotaan, dan profil jasa travel yang bersangkutan.
- User pengunjung merupakan individu, bisa siapa saja yang menginginkan mengunjungi situs untuk memperoleh informasi, selama mempunyai telepon genggam dengan fasilitas WAP.

b. User Anggota

- User anggota merupakan individu, bisa siapa saja yang mampu melakukan pemesanan jasa travel, selama mempunyai telepon genggam dengan fasilitas WAP.

c. Mekanisme Pendaftaran Anggota

- User pengunjung bisa mendapatkan informasi yang lengkap mengenai pemesanan jasa travel. Adapun informasi yang diberikan adalah sebagai berikut:
 - Informasi harga pemesanan jasa, meliputi: kota tujuan, kelas kendaraan, harga travel per kota tujuan, dan cara pembatalan pemesanan.
 - Persyaratan keanggotaan
 - Profil perusahaan jasa travel, meliputi: alamat perusahaan, nomor telepon travel, dan Penanggung Jawab Travel.
- User pengunjung harus melakukan registrasi untuk dapat melakukan pemesanan jasa travel dengan cara melakukan registrasi pendaftaran menggunakan telepon genggam dengan mengisi data-data yang dibutuhkan, yaitu: *nick user*, *password*, nama, alamat, nomor KTP dan nomor telepon. Setelah mengisi data maka *nick user* dan *password* yang telah didaftarkan dapat digunakan untuk login sebelum dipersilahkan melakukan pemesanan jasa.
- Apabila user pengunjung tidak dapat melakukan registrasi anggota dikarenakan adanya masalah pada sistem, user pengunjung dapat menghubungi admin untuk didaftarkan menjadi anggota.

d. Mekanisme Pemesanan Jasa

- Untuk melakukan pemesanan Jasa Travel, pertama kali anggota harus melakukan *login* dengan memasukkan *nick user* dan *password* yang telah terdaftar dalam sistem, dan memilih *link* Pesan Jasa apabila *login* yang dilakukan valid. Proses ini secara berurutan adalah memilih jadwal keberangkatan, kota tujuan, kelas kendaraan dan jumlah penumpang.

e. Mekanisme Pembatalan Pemesanan Jasa

- Untuk melakukan pembatalan pemesanan jasa, pertama kali anggota harus melakukan *login* dengan memasukkan *nick user* dan *password* yang telah terdaftar dalam sistem, dan memilih *link* Batal Jasa apabila *login* yang dilakukan valid.
- Selanjutnya user anggota akan ditampilkan kode transaksi beserta keterangan dari jasa travel yang telah dipesan sebelumnya yang masih berlaku. User anggota memilih kode transaksi yang akan dibatalkan.

f. User Manajemen (Pengelola Perusahaan)

- Pengelola perusahaan berhak melarang atau mencegah data-data yang akan diberikan kepada pengunjung situs.
- Pengelola perusahaan mengontrol laporan perusahaan melalui *Internet*.

g. User Customer Service (Pengelola Data)

- Pengelola data bertanggung jawab terhadap data-data yang akan ditampilkan kepada pengunjung situs.
- Pengelola data harus mengikuti aturan-aturan yang telah diberikan oleh pengelola perusahaan.
- Pengelola data merupakan individu, petugas yang berusia di atas 18 tahun dan ditugaskan pengelola perusahaan untuk mengelola data.
- Pengelola data berhak mengelola data, yaitu: menambah data, meng-*update* data, atau menghapus data melalui *Internet*, sesuai yang ditugaskan oleh pengelola perusahaan.

4.1.3 Diagram Alir Sistem Pemesanan Travel

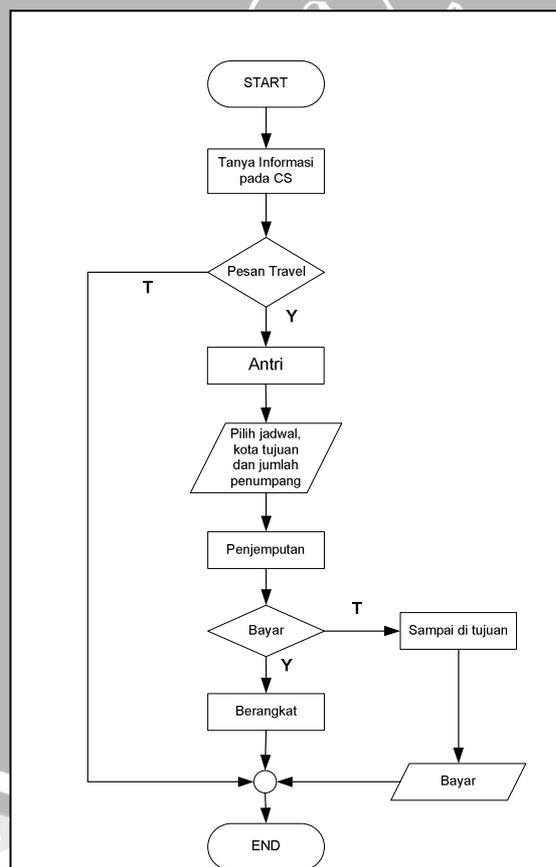
Diagram Alir (*flowchart*) merupakan metode untuk menggambarkan tahap-tahap pemecahan masalah dengan merepresentasikan simbol-simbol tertentu yang mudah dimengerti, mudah digunakan, dan standar. Tujuan utama penggunaan *flowchart* adalah

untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi, dan jelas dengan menggunakan simbol-simbol yang standar.

Diagram alir sistem pemesanan travel ini digunakan untuk menunjukkan perbedaan antara sistem pemesanan jasa travel secara manual, dan sistem pemesanan jasa travel menggunakan telepon genggam dengan fasilitas WAP yang akan dirancang dengan tujuan agar lebih mudah dipahami.

4.1.3.1 Perbedaan Sistem Pemesanan Travel Secara Manual dan WAP

Berikut ini merupakan perbedaan antara sistem pemesanan jasa travel secara manual dan sistem pemesanan jasa travel menggunakan telepon genggam dengan fasilitas WAP. Pada sistem secara manual, pelanggan datang ke kantor jasa travel atau menghubungi pihak *customer service* untuk melakukan pemesanan, pembatalan, maupun pembayaran pemesanan. Diagram alir sistem secara manual ditunjukkan pada Gambar 4.2.

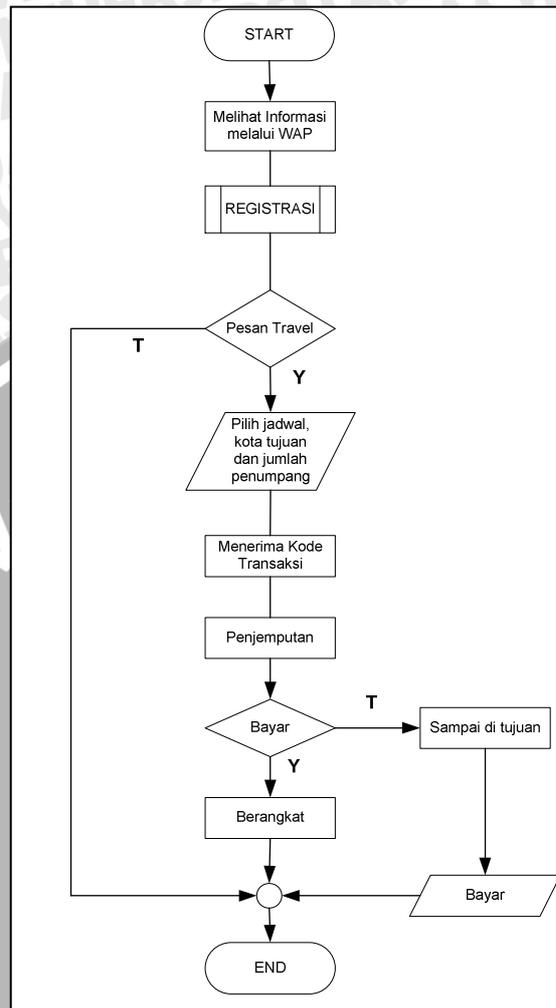


Gambar 4.2 Diagram Alir Pemesanan Travel secara Manual

Sumber : Analisis dan Perancangan

Pada sistem yang menggunakan telepon genggam dengan fasilitas WAP, pelanggan tidak harus datang ke kantor jasa travel ataupun menelepon *customer service*

dalam melakukan pemesanan, pembatalan, dan pembayaran jasa. Diagram alir sistem menggunakan telepon genggam dengan fasilitas WAP ditunjukkan pada Gambar 4.3.



Gambar 4.3 Diagram Alir Sistem Pemesanan Travel dengan WAP

Sumber : Analisis dan Perancangan

4.1.4 Analisis Kebutuhan Perangkat Lunak

Untuk mendapatkan informasi dan memesan jasa travel yang dapat diakses melalui Internet, akan dirancang dan dibuat aplikasi penyedia layanan pemesanan jasa travel dan informasi jasa travel berbasis *web*. Aplikasi ini dapat diakses secara *online* melalui *web* menggunakan telepon genggam yang mempunyai fasilitas layanan WAP dan terdapat jaringan GPRS.

Dari hasil observasi yang dilakukan melalui pengumpulan data penunjang untuk proses pembangunan sistem, survey telah dilakukan di salah satu kantor penyedia jasa travel di malang yaitu Travel Jasa Tirta dan diketahui bahwa sampai saat ini sistem pemesanan, pembatalan, dan pembayaran jasa pada Travel Jasa Tirta masih

menggunakan cara manual, yaitu datang langsung atau melalui telepon ke kantor yang bersangkutan. Berdasarkan hasil survey tersebut, maka untuk menciptakan alternatif sistem pemesanan, pembatalan, dan pembayaran jasa travel secara *online*, didapatkan kebutuhan yang secara spesifik dijelaskan sebagai berikut:

1. Semua *user* (kecuali pengunjung) harus melakukan *login* untuk dapat mengakses sistem.
2. *User* dibagi menjadi dua bagian:
 - a. *User* yang mengakses sistem pemesanan jasa travel ini melalui *interface web*, yaitu admin perusahaan, manajemen, dan admin perangkat lunak.
 - b. *User* yang mengakses sistem pemesanan jasa travel ini melalui *interface WAP*, yaitu anggota dan pengunjung.
3. User Customer Service dapat memasukkan data jadwal keberangkatan, data kendaraan, dan data kota tujuan. Data kendaraan yang dimasukkan berupa kode kendaraan, kapasitas kendaraan dan harga pemesanan jasa travel.
User Admin (Admin Perangkat Lunak) dapat memperoleh informasi anggota, melakukan registrasi untuk anggota apabila diperlukan, memblokir *user*, dan mengganti *password user* (anggota, admin perusahaan, manajemen).
4. Kepala Jasa Travel selaku manajemen dapat mengetahui informasi kendaraan, informasi anggota, laporan keberangkatan, laporan pemesanan, laporan pembatalan, dan laporan pembayaran pemesanan jasa.
5. Anggota dapat melakukan pemesanan beserta pembayaran pemesanan atau pembatalan pemesanan yang telah dipesan sebelumnya. Anggota juga dapat memperoleh informasi pemesanan yang telah dipesan oleh anggota bersangkutan yang masih berlaku, yaitu: tanggal pesan, jam pesan, kota tujuan, alamat anggota, alamat tujuan dan nama anggota.
6. Pengunjung dapat melihat informasi harga pemesanan jasa, persyaratan anggota, dan profil Jasa Travel yang bersangkutan. Apabila akan melakukan pemesanan jasa, maka pengunjung harus melakukan registrasi untuk menjadi anggota terlebih dahulu dan akan mendapatkan *user id* dan *password*.
7. Implementasi sistem ini dibuat dengan bahasa pemrograman web *server-side* yaitu PHP dengan *database MySQL* serta Apache *Web Server* sebagai *server*,
8. Sistem yang dibuat menggunakan *database* yang sudah dinormalisasi sebelumnya.
9. Sistem operasi yang digunakan adalah sistem operasi WindowsXP Professional Edition SP2.

10. Pengaksesan sistem pemesanan jasa travel oleh anggota dilakukan dengan *browsing* menggunakan telepon genggam yang mempunyai fasilitas layanan WAP dan terdapat jaringan GPRS.
11. Sistem yang dibuat harus terdapat *site map*.

4.1.4.1 Spesifikasi Pengguna Perangkat Lunak

Perangkat lunak tidak lepas dari interaksi pengguna (*user*) dan sistem. Pada sistem ini, *user* dapat digolongkan menjadi lima macam, yaitu: pengunjung, anggota, admin perusahaan, admin perangkat lunak dan manajemen. Kelima macam *user* tersebut mempunyai hak akses yang berbeda. Hak akses dari tiap *user* dapat dijelaskan dengan definisi dan spesifikasi sebagai berikut:

A. Pengunjung

Pengunjung adalah semua pengguna yang membuka aplikasi ini, tetapi hanya memiliki hak untuk mendapatkan informasi yang telah disediakan oleh aplikasi, sehingga pengguna biasa tidak perlu melakukan login untuk mengakses aplikasi. Hak akses yang diberikan adalah sebagai berikut:

Definisi:

1. Pengunjung dapat mengakses Sistem Informasi Jasa Travel dan bila akan memesan jasa travel harus registrasi menjadi anggota terlebih dahulu. (SPT_010)

Spesifikasi:

- 1.1 Pengunjung hanya dapat memperoleh informasi yang disediakan oleh admin yang meliputi: informasi jadwal keberangkatan, kota tujuan, informasi harga, syarat-syarat pemesanan jasa travel dan profil Jasa Travel yang bersangkutan. (SPT_011)
- 1.2 Pengunjung harus melakukan pendaftaran / registrasi terlebih dahulu untuk mendapatkan *user id* dan *password* yang dapat digunakan untuk memesan jasa. (SPT_012)
- 1.3 Apabila pengunjung mengalami masalah saat melakukan registrasi, pengunjung dapat menghubungi admin untuk didaftarkan sebagai anggota. (SPT_013)

B. Anggota

Anggota adalah pengguna yang mempunyai hak untuk memesan jasa travel melalui WAP, sehingga diperlukan pendaftaran / registrasi terlebih dahulu sebelum bisa memesan jasa travel. Untuk lebih jelasnya dapat dijelaskan sebagai berikut:

Definisi:

2. Anggota harus melakukan *login* terlebih dahulu untuk dapat memesan jasa travel. (SPT_020)

Spesifikasi:

- 2.1 Anggota memasukkan *user id* dan *password* untuk *login* (SPT_021)
- 2.2 Anggota dapat melanjutkan untuk melakukan pemesanan jasa travel, pembatalan pemesanan jasa travel, atau mendapatkan informasi pemesanan jasa travel jika *login* yang diberikan adalah valid. (SPT_022)
- 2.3 Sistem akan memberikan pesan kesalahan saat *login* yang diberikan tidak valid dan yang bersangkutan akan berstatus sebagai pengunjung (SPT_023)
- 2.4 Anggota dapat memperoleh informasi dan kode transaksi pemesanan jasa travel anggota bersangkutan. (SPT_024)
- 2.5 Anggota dapat melakukan pembatalan pemesanan jasa travel dengan memilih kode transaksi pemesanan yang telah didapatkan. (SPT_025)

C. Customer Service

Customer Service adalah pengguna yang mempunyai hak dalam memasukkan data-data yang berhubungan dengan pemesanan jasa travel pada aplikasi ini dan pengguna yang mempunyai hak dalam hal yang berhubungan dengan *user* pada aplikasi ini, sehingga diperlukan login terlebih dahulu sebelum mendapatkan fasilitas-fasilitas customer service.

Definisi:

3. Customer Service harus dapat mengolah data yaitu menambah, menghapus dan mengubah data yang dilakukan melalui *web* dan juga harus dapat mengolah data yang berhubungan dengan *user*, yaitu menambah, menghapus, mengubah, ataupun memblokir *user*, yang dilakukan melalui *web* (SPT_030).

Spesifikasi:

- 3.1 Customer Service harus memasukkan login berupa *user id* dan *password* terlebih dahulu sebelum mendapatkan fasilitas-fasilitas Customer Service. (SPT_032)

- 3.2 Customer Service harus dapat menyediakan informasi pemesanan jasa travel yang meliputi: jadwal keberangkatan, tarif travel, data-data kendaraan, dan kota tujuan. (SPT_033)
- 3.3 Customer Service harus dapat melakukan *update* data dengan mengubah status kendaraan dan keberangkatan (SPT_034)
- 3.4 Customer Service dapat mengubah, menambah, atau menghapus informasi pengguna yang meliputi: data pemesanan anggota, data pembatalan pemesanan anggota, dan data pembayaran anggota. (SPT_035)
- 3.5 Customer Service dapat mendaftarkan anggota, mengubah, atau menghapus data – data anggota apabila diperlukan. (SPT_036)

D. Manajemen

Manajemen adalah pengelola perusahaan (*manager*) Jasa Travel.

Definisi:

4. Manajemen mempunyai kebijakan untuk mengontrol perkembangan perusahaan yaitu menerima hasil laporan dari pengguna aplikasi, yang dilakukan melalui *web*. (SPT_040)

Spesifikasi:

- 4.1 Manajemen harus memasukkan login berupa *user id* dan *password* terlebih dahulu sebelum mendapatkan fasilitas-fasilitas manajemen. (SPT_041)
- 4.2 Manajemen harus menerima hasil laporan dari pengguna aplikasi yaitu laporan identitas anggota, data pemesanan, pembatalan, dan pembayaran pemesanan, dan jumlah penumpang yang tersisa (SPT_042).

4.1.4.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak merupakan hal-hal yang dapat dilakukan oleh perangkat lunak terhadap *user* ataupun perangkat lunak itu sendiri dan kebutuhan perangkat lunak agar dapat diakses oleh *user*.

Definisi:

6. Pengaksesan informasi aplikasi oleh anggota dan pengunjung dilakukan dengan *browsing* ke Internet menggunakan telepon genggam. (SPT_060)

Spesifikasi:

- 6.1 Pengaksesan informasi aplikasi oleh anggota dan pengunjung dilakukan dengan *browsing* menggunakan telepon genggam (handphone) yang memiliki fasilitas layanan WAP dan terdapat jaringan GPRS. (SPT_051)

Definisi:

7. Pengaksesan informasi aplikasi oleh admin perusahaan, dan manajemen dilakukan dengan *browsing* ke Internet melalui *web*. (SPT_070)

Spesifikasi:

- 7.1 Pengaksesan informasi aplikasi oleh admin perusahaan, dan manajemen dilakukan dengan *browsing* ke Internet melalui *web*. (SPT_071)

Definisi:

8. Sistem yang dibuat harus menggunakan *database* yang akan dirancang dan dinormalisasi sebelumnya. (SPT_080)

Spesifikasi:

- 8.1 Sistem yang dibuat harus menggunakan *database* yang akan dirancang dan dilakukan normalisasi tabel. (SPT_081)

Definisi:

9. Sistem diimplementasikan dengan bahasa pemrograman PHP dan WML, serta basis data MySQL. (SPT_090)

Spesifikasi:

- 9.1 Sistem diimplementasikan menggunakan bahasa pemrograman PHP sebagai penghubung antara sistem dengan *database*. (SPT_091)
- 9.2 Sistem diimplementasikan menggunakan bahasa pemrograman WML untuk dapat memberikan tampilan pada layar telepon genggam. (SPT_092)
- 9.3 Sistem diimplementasikan menggunakan *database* MySQL sebagai sarana penyimpanan data. (SPT_093)

Definisi:

10. Sistem operasi yang digunakan adalah sistem operasi Windows. (SPT_100)

Spesifikasi:

- 10.1 Sistem yang akan dibuat menggunakan Microsoft Windows XP Professional Edition SP 2. (SPT_101)

Definisi:

11. Dokumentasi sistem yang dibuat harus terdapat *site map*. (SPT_110)

Spesifikasi:

11.1 Site map merupakan penjelasan dari menu yang disediakan oleh sistem, pada *Pengunjung* meliputi: registrasi anggota untuk menjadi anggota, informasi tarif travel, informasi persyaratan keanggotaan, dan profil perusahaan Jasa Travel bersangkutan. Pada *Anggota* meliputi: pemesanan jasa travel, pembatalan jasa travel, dan informasi transaksi. Pada *Admin Perusahaan* meliputi: memasukkan dan mengedit data kendaraan, kota tujuan, tarif travel dan jadwal keberangkatan. Pada *Admin Perangkat Lunak* meliputi: memasukkan dan mengedit data anggota, memblokir anggota, mengganti *password* anggota, admin perusahaan, dan manajemen. Pada *Manajemen* meliputi: pilihan untuk menampilkan laporan anggota, kendaraan, keberangkatan, pemesanan, pembayaran, dan pembatalan (SPT_101).

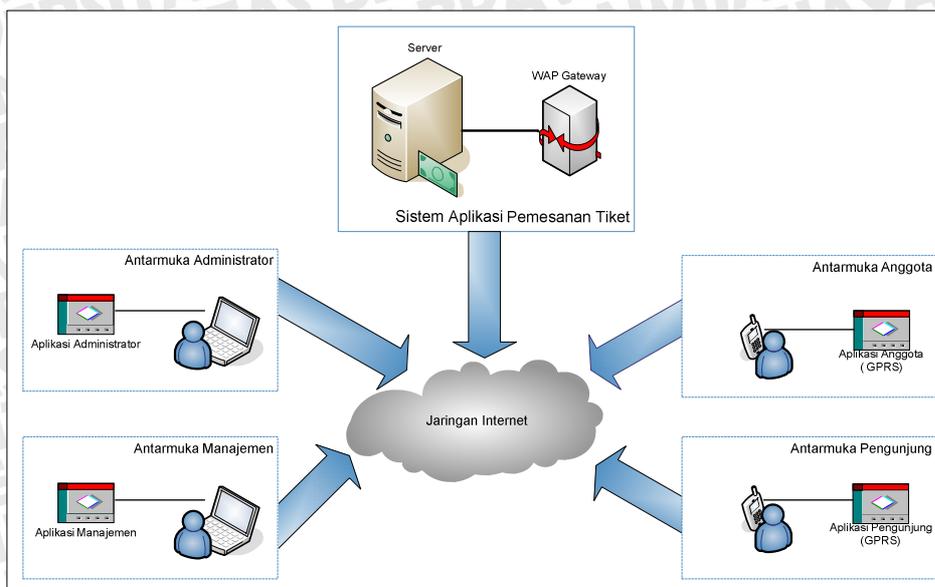
4.2 Perancangan Perangkat Lunak

Berdasarkan data dan informasi sebelumnya, perancangan Sistem Pemesanan Jasa Travel Berbasis WAP ini meliputi perancangan:

1. Diagram Blok Sistem
2. DFD
3. Basis data
4. Sistem *Site Map*
5. *Antarmuka User*

4.2.1 Diagram Blok Sistem

Diagram blok sistem menggambarkan setiap blok atau bagian dari sistem aplikasi. Aplikasi pemesanan Jasa Travel ini dirancang untuk dapat dijalankan secara *online*, seperti yang ditunjukkan pada Gambar 4.4



Gambar 4.4 Diagram Blok Sistem Aplikasi Sistem Pemesanan Jasa Travel Secara Online
 Sumber : Analisis dan Perancangan

Diagram blok sistem aplikasi ini terdiri dari lima blok yang terhubung melalui jaringan Internet. Masing-masing blok dapat dijelaskan sebagai berikut:

- **Sistem Aplikasi Pemesanan Travel**
 Merupakan bagian inti dari aplikasi ini. Pada bagian ini *server* terhubung dengan *WAP Gateway* sebagai semacam penerjemah informasi dari *server* sebelum diteruskan kepada pengguna Anggota dan Pengunjung dengan menggunakan telepon genggam.
- **Antarmuka Anggota**
 Antarmuka Anggota adalah antarmuka yang digunakan oleh anggota (*user* Pengunjung yang telah melakukan registrasi) untuk mengakses aplikasi ini. Akses yang dilakukan adalah pemesanan atau pembatalan Jasa Travel, dan informasi transaksi anggota. Aplikasi ini diakses melalui telepon genggam dengan menggunakan jaringan GPRS.
- **Antarmuka Pengunjung**
 Antarmuka Pengunjung adalah aplikasi yang sebenarnya sama dengan Antarmuka Anggota dalam mengakses aplikasi ini. Yang membedakan adalah Pengunjung merupakan *user* yang belum melakukan registrasi menjadi anggota. Akses yang didapat dari Antarmuka Pengunjung adalah informasi tentang tarif, jadwal keberangkatan, kota tujuan, profil, dan persyaratan keanggotaan.

- Antarmuka Administrator

Antarmuka Administrator adalah antarmuka yang digunakan oleh administrator untuk melakukan manajemen data pada aplikasi ini. Akses ini diakses langsung ke server tanpa melalui *WAP Gateway*.

- Antarmuka Manajemen

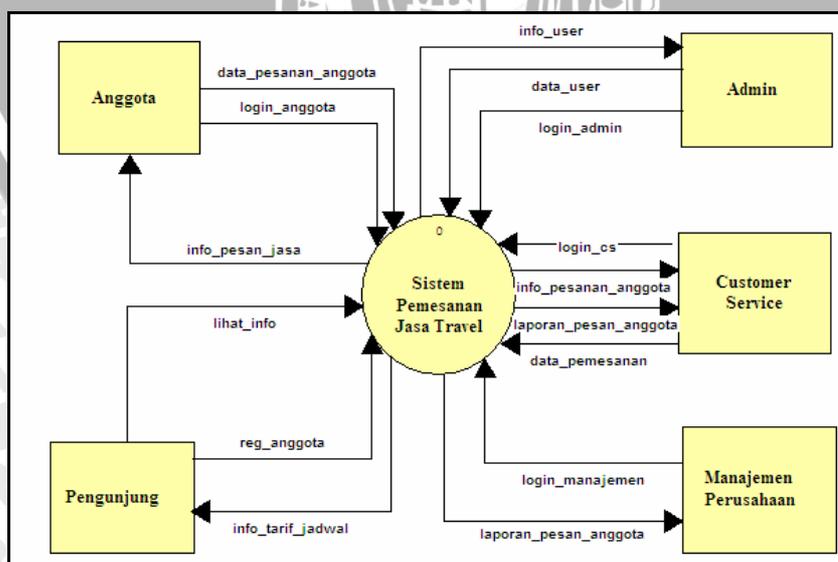
Antarmuka Manajemen adalah antarmuka yang digunakan oleh pihak manajemen untuk memeriksa data pada aplikasi ini, yang ditampilkan dalam bentuk laporan. Akses ini juga diakses langsung ke server tanpa melalui *WAP Gateway*.

4.2.2 Data Flow Diagram

Data Flow Diagram (DFD) adalah suatu diagram yang menggambarkan model komponen sistem yang akan dibuat, dalam hal ini aplikasi sistem informasi jasa travel berbasis WAP.

4.2.2.1 Context Diagram

Context Diagram menjelaskan hubungan sistem dengan lingkungan atau kesatuan luar. Pada sistem ini, diagram konteks yang dibuat melibatkan empat kesatuan luar, yaitu : Anggota, Pengunjung, Admin Perusahaan, *Customer Service*, dan Manajemen. *Context Diagram* direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem. DFD yang pertama kali dibuat adalah DFD level 0 atau yang sering disebut *Context Diagram*. *Context Diagram* Sistem Pemesanan Jasa Travel ditunjukkan pada Gambar 4.5 berikut:



Gambar 4.5 Context Diagram Aplikasi Sistem Pemesanan Jasa Travel
 Sumber : Analisis dan Perancangan

Berdasarkan Gambar 4.5 proses “Sistem Pemesanan Jasa Travel” mempunyai data masukan berupa:

1. *login_anggota* [IRS_0110]

login_anggota oleh *user* (anggota) untuk dilakukan proses validasi login.

Parameter yang digunakan adalah:

a. *user_anggota* [IRS_0111]

Berisi masukan *nama* yang diberikan oleh anggota.

d. *pass_anggota* [IRS_0113]

Berisi masukan *pass* yang diberikan oleh anggota.

2. *login_cs* [IRS_0120]

login_cs diberikan oleh *user* (admin perusahaan) untuk dilakukan proses validasi login.

Parameter yang digunakan adalah:

a. *user_cs* [IRS_0121]

Berisi masukan *username* yang diberikan oleh admin.

b. *pass_cs* [IRS_0122]

Berisi masukan *password* yang diberikan oleh admin.

3. *reg_anggota* [IRS_0130]

reg_anggota diberikan oleh *user pengunjung*. Pengunjung adalah semua pengguna yang membuka aplikasi ini, tetapi hanya memiliki hak untuk mendapatkan informasi yang telah disediakan pada tampilan awal aplikasi, dan perlu melakukan pendaftaran / login untuk mengakses aplikasi, yaitu memesan atau membatalkan pemesanan.

Pengunjung dapat melakukan registrasi dengan memasukkan data-data sebagai berikut :

a. *username* [IRS_0131]

Merupakan *username* yang diisi oleh calon anggota yang bersangkutan yang selanjutnya akan digunakan oleh anggota untuk mengisi *login*.

b. *password* [IRS_0132]

Merupakan *password* yang diisi oleh calon anggota yang bersangkutan yang selanjutnya akan digunakan oleh anggota untuk mengisi *login*.

c. *nama_lengkap* [IRS_0133]

Merupakan nama lengkap calon anggota yang selanjutnya akan digunakan sebagai data anggota.

d. *alamat* [IRS_0134]

Merupakan alamat tempat tinggal calon anggota yang selanjutnya akan digunakan sebagai data anggota.

e. *no_ktp* [IRS_0135]

Merupakan nomor tanda pengenal calon anggota yang selanjutnya akan digunakan sebagai data anggota.

f. *no_telp* [IRS_0136]

Merupakan nomor telepon atau *handphone* calon anggota yang selanjutnya akan digunakan sebagai data anggota.

4. *login_manajemen* [IRS_0140]

login_manajemen diberikan oleh *user* (manajemen) untuk dilakukan proses validasi login. Manajemen adalah pengguna yang mempunyai hak *manage* aplikasi.

Parameter yang digunakan adalah :

a. *user_manajemen* [IRS_0141]

Berisi masukan *username* yang diberikan oleh manajemen.

b. *pass_manajemen* [IRS_0142]

Berisi masukan *password* yang diberikan oleh manajemen.

5. *data_pemesanan* [IRS_0150]

data_pemesanan diberikan oleh *user* (customer service). Customer service memasukkan data / informasi yang berhubungan dengan pemesanan travel.

Parameter yang digunakan adalah:

a. *jadwal_keberangkatan* [IRS_0151]

Berisi masukan jadwal keberangkatan travel yang dapat dipilih oleh anggota dalam melakukan pemesanan travel.

b. *kota_tujuan* [IRS_0152]

Berisi masukan kota tujuan yang dapat dipilih oleh anggota dalam melakukan pemesanan travel.

c. *jenis_layanan* [IRS_0153]

Berisi masukan jenis layanan baik jasa travel ataupun kurir yang dapat dipilih oleh anggota dalam melakukan pemesanan travel.

d. *tarif_travel* [IRS_0154]

Berisi masukan tarif dari jenis layanan travel yang akan digunakan oleh pemesan travel.

6. *data_pesan_anggota* [IRS_0160]

data_pesan_anggota diberikan oleh anggota. Anggota memasukkan data tentang pemesanan jasa travel.

Parameter yang digunakan adalah:

a. *waktu_berangkat* [IRS_0161]

Berisi masukan waktu yang dipilih anggota dalam pemesanan travel.

b. *jenis* [IRS_0162]

Berisi masukan jenis layanan yang dipilih anggota dalam pemesanan travel.

c. *kota_tujuan* [IRS_0163]

Berisi masukan kota tujuan yang dipilih anggota dalam pemesanan travel.

d. *jml_penumpang* [IRS_0164]

Berisi masukan jumlah penumpang yang dipesan oleh anggota.

e. *tanggal_berangkat* [IRS_0165]

Berisi masukan tanggal yang dipilih anggota dalam pemesanan travel.

7. *lihat_info* [IRS_0170]

lihat_info diberikan oleh pengunjung. Pengunjung dapat mendapatkan informasi yang telah disediakan untuk pengunjung tanpa harus *login* sebagai anggota.

8. *login_admin* [IRS_0180]

login_admin diberikan oleh *user* (admin perusahaan) itu sendiri untuk dilakukan proses validasi login. Admin perusahaan adalah pengguna yang mempunyai hak mengubah data yang berhubungan dengan hak akses *user*.

Parameter yang digunakan adalah :

a. *user_admin* [IRS_0181]

Berisi masukan *username* yang diberikan oleh admin perusahaan.

b. *pass_admin* [IRS_0182]

Berisi masukan *password* yang diberikan oleh admin perusahaan.

Berdasarkan Gambar 4.5 proses “Sistem Pemesanan Pemesanan Jasa travel” mempunyai data keluaran berupa:

1. *lihat_pesanan* [IRS_1000]

Berisi data-data transaksi pemesanan travel anggota bersangkutan yang masih berlaku, meliputi : tanggal keberangkatan, waktu keberangkatan, jenis layanan, dan kota tujuan.

2. *jadwal_keberangkatan* [IRS_1100]

Berisi informasi harga pemesanan travel berdasarkan kota tujuan dan jenis jasa yang dipesan yang dapat dilihat oleh pengunjung tanpa harus *login* sebagai anggota.

4. *laporan_pesan_anggota* [IRS_2400]

Berisi laporan pemesanan travel anggota untuk pihak manajemen, yang meliputi : laporan data anggota, laporan data mobil, laporan jadwal keberangkatan, laporan data pembayaran, dan laporan data pembatalan.

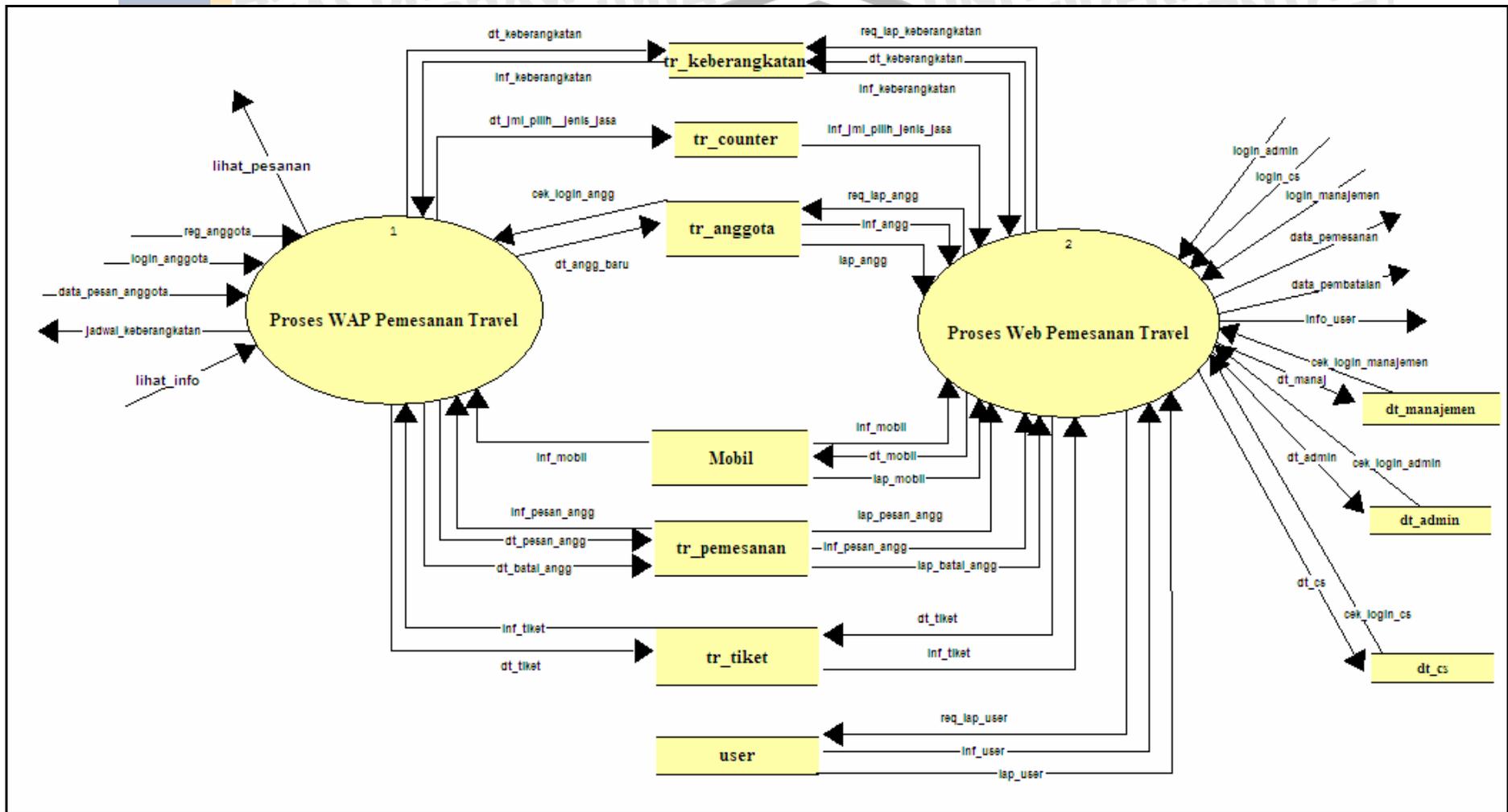
5. *info_user* [IRS_2500]

Berisi informasi tentang *user* pada sistem ini untuk pihak admin perusahaan, yang meliputi: data anggota, data *user*, data administrator, dan data perusahaan.

Untuk masuk ke dalam sistem tersebut *user* anggota, admin perusahaan dan manajemen harus memasukkan login terlebih dahulu. Sedangkan *user* pengunjung tidak perlu memasukkan login untuk masuk ke dalam sistem. Jika *login_anggota*, *login_admin*, *login_admin* dan *login_manajemen* yang dimasukan sudah benar maka *user* dapat menuju ke proses berikutnya. Jika *user* salah dalam memasukan *login*, maka proses akan tetap berada pada proses *login*.

4.2.2.2 DFD Level 1

DFD level 1 menggambarkan proses-proses yang lebih detail dari diagram konteks untuk Sistem Pemesanan Pemesanan travel Jasa travel. DFD level 1 digambarkan pada Gambar 4.6.



Gambar 4.6 DFD Level 1 Sistem Pemesanan Jasa Travel Dengan Layanan WAP

Gambar 4.6 merupakan DFD Level 1 dari Sistem Pemesanan Jasa travel dimana sistem tersebut dibagi menjadi 2 proses, yaitu : Proses WAP Pemesanan Travel dan Proses Web Pemesanan Travel. Dari diagram tersebut, maka dapat dijabarkan lebih detail proses-proses tersebut sebagai berikut:

1. Proses WAP Pemesanan Travel (Proses 1.1)

Merupakan proses yang terjadi dengan pengaksesan WAP pada sistem pemesanan jasa travel dan kurir. *User* yang mengakses WAP adalah anggota dan pengunjung.

Proses ini mempunyai data masukan :

a. *login_anggota* [IRS_0110]

b. *reg_anggota* [IRS_0130]

c. *lihat_info* [IRS_0170]

d. *data_pesan_anggota* [IRS_0160]

e. *cek_login_angg* [IRS_0230]

cek_login_angg berisi data login yang tersimpan dalam *data store dt_anggota* yang digunakan untuk validasi login anggota.

f. *inf_keberangkatan* [IRS_0190]

inf_keberangkatan adalah informasi tentang jadwal keberangkatan yang dapat dipilih anggota, berupa rincian jadwal keberangkatan yaitu kota tujuan, hari keberangkatan, jam berangkat beserta tarif untuk tiap jenis pemesanan jasa.

g. *inf_mobil* [IRS_0200]

inf_mobil adalah informasi kendaraan yang akan digunakan untuk jadwal keberangkatan yang telah dipilih oleh anggota, meliputi : jenis kendaraan dan jumlah penumpang.

h. *inf_pesan_angg* [IRS_0210]

inf_pesan_angg adalah informasi tentang pemesanan jasa travel yang telah dimasukkan oleh anggota, yang meliputi: jenis layanan, tanggal pemesanan, jumlah pemesanan, jam pemesanan, dan harga total.

i. *inf_tiket* [IRS_0220]

inf_tiket adalah informasi tentang sisa tiket yang dapat dipesan oleh anggota pada tiap jenis pemesanan jasa.

Proses ini mempunyai data keluaran :

a. *lihat_pesanan* [IRS_1000]

lihat_pesanan adalah informasi pemesanan travel anggota bersangkutan meliputi: jenis jasa, tanggal keberangkatan, waktu keberangkatan, kota tujuan, alamat tujuan dan jumlah penumpang atau keterangan barang.

b. *jadwal_keberangkatan* [IRS_1100]

jadwal_keberangkatan adalah jadwal berangkat untuk travel dan kurir.

c. *dt_angg_baru* [IRS_1400]

dt_angg_baru adalah data pengunjung yang telah terdaftar menjadi anggota.

d. *dt_jml_pilih_jenis_jasa* [IRS_1300]

dt_jml_pilih_jenis_jasa adalah data jumlah total jenis pesanan per jadwal keberangkatan yang dipilih oleh user anggota.

e. *dt_pesan_angg* [IRS_1500]

dt_pesan_angg adalah data tentang pemesanan jasa travel yang dimasukkan anggota.

f. *dt_batal_angg* [IRS_1600]

dt_batal_angg adalah data tentang pembatalan pemesanan jasa travel yang dimasukkan anggota sesuai dengan kode transaksi pemesanan.

g. *dt_tiket* [IRS_1700]

dt_tiket adalah data sisa tiket yang didapat setelah dikurangi data pemesanan yang dilakukan oleh anggota berdasarkan jenis jasa.

Proses ini menggunakan *data store* :

a. *tr_anggota* [IRS_0001]

b. *tr_mobil* [IRS_0002]

c. *tr_counter* [IRS_0003]

d. *tr_keberangkatan* [IRS_0004]

e. *tr_pemesanan* [IRS_0005]

f. *tr_tiket* [IRS_0006]

g. *user* [IRS_0007]

2. Proses *Web* Pemesanan Pemesanan travel (proses 1.2)

Merupakan proses yang terjadi dengan pengaksesan *Web* pada sistem pemesanan jasa travel. *User* yang mengakses *Web* adalah admin, customer service, dan manajemen.

Proses ini mempunyai data masukan :

a. *login_admin* [IRS_0180]

b. *login_cs* [IRS_0120]

c. *login_manajemen* [IRS_0140]

d. *cek_login_admin* [IRS_0240]

cek_login_admin berisi data *login* yang digunakan untuk validasi *login* admin.

e. *cek_login_manajemen* [IRS_0230]

cek_login_manajemen berisi data *login* yang tersimpan dalam *data store* yang digunakan untuk validasi *login* manajemen.

f. *cek_login_cs* [IRS_0250]

cek_login_cs berisi data *login* yang tersimpan dalam *data store* untuk validasi *login* customer service.

g. *inf_angg* [IRS_0280]

inf_angg adalah informasi tentang anggota yang sudah terdaftar dan terdapat di *data store*.

h. *inf_keberangkatan* [IRS_0260]

inf_keberangkatan adalah informasi tentang jadwal keberangkatan yang terdapat di *data store*.

i. *inf_jml_pilih_jenis_jasa* [IRS_0270]

inf_jml_pilih_jenis_jasa adalah informasi tentang jumlah pemesanan kurir atau travel yang dilakukan oleh anggota.

j. *inf_mobil* [IRS_0300]

inf_mobil adalah informasi tentang armada mobil yang digunakan yang terdapat di *data store mobil*.

k. *lap_mobil* [IRS_0310]

lap_mobil adalah informasi tentang laporan rinci dari armada mobil yang digunakan oleh pihak travel untuk *user* manajemen.

l. *lap_angg* [IRS_0290]

lap_angg adalah laporan tentang identitas dari anggota, berasal dari *data store dt_anggota*, yang ditujukan untuk *user* manajemen.

- m. *lap_pesan_anggota* [IRS_0320]
lap_pesan_anggota adalah laporan tentang pemesanan jasa oleh anggota, yang ditujukan untuk *user* manajemen.
- n. *lap_batal_angg* [IRS_0330]
lap_batal_angg adalah laporan tentang pembatalan pemesanan jasa oleh anggota berdasarkan kode transaksi, yang ditujukan untuk *user* manajemen.
- o. *inf_pesan_angg* [IRS_0340]
inf_pesan_angg adalah informasi tentang detail pemesanan jasa yang dilakukan oleh anggota.
- p. *inf_tiket* [IRS_0350]
inf_tiket adalah *inf_tiket* adalah informasi tentang sisa tiket yang masih tersisa pada tiap jenis pemesanan jasa.
- q. *info_user* [IRS_1800]
info_user adalah informasi tentang *user* yang terdaftar pada sistem pemesanan jasa travel .
- r. *lap_user* [IRS_0370]
lap_user adalah laporan tentang *user* yang terdaftar pada sistem pemesanan jasa travel. Ditujukan untuk *user* manajemen.

\
Proses ini mempunyai data keluaran :

- a. *data_pemesanan* [IRS_0150]
data_pemesanan adalah informasi pemesanan jasa oleh anggota yang meliputi : nomor transaksi beserta informasi transaksi, dan identitas anggota bersangkutan dan ditujukan untuk *user* manajemen.
- b. *data_pembatalan* [IRS_0380]
data_pembatalan adalah informasi pembatalan pemesanan jasa oleh anggota, ditujukan untuk *user* manajemen.
- c. *info_user* [IRS_1800]
- d. *req_lap_keberangkatan* [IRS_1900]
req_lap_keberangkatan adalah permintaan dari *user* manajemen untuk menampilkan data laporan jadwal keberangkatan travel pada hari ini.
- e. *dt_keberangkatan* [IRS_2000]
dt_keberangkatan adalah data jadwal keberangkatan travel yang di *update* oleh *customer service*..

f. *req_lap_angg* [IRS_2100]

req_lap_angg adalah permintaan dari *user* manajemen untuk menampilkan data rinci dari *user* anggota yang telah terdaftar.

g. *dt_mobil* [IRS_2200]

dt_mobil adalah data-data tentang mobil yang akan digunakan dan dimasukkan oleh customer service ke data store.

h. *dt_tiket* [IRS_2300]

dt_tiket adalah data-data tentang kapasitas penumpang dan jumlah penumpang maksimal yang dapat dipesan dan dimasukkan oleh customer service ke data store.

i. *req_lap_mobil* [IRS_2500]

req_lap_mobil adalah permintaan dari *user* manajemen untuk menampilkan laporan armada travel..

j. *req_lap_user* [IRS_2400]

req_lap_user adalah permintaan dari *user* manajemen untuk menampilkan siapa saja *user* yang terdaftar dalam sistem.

Proses ini menggunakan *data store* :

h. *tr_anggota* [IRS_0001]

i. *tr_mobil* [IRS_0002]

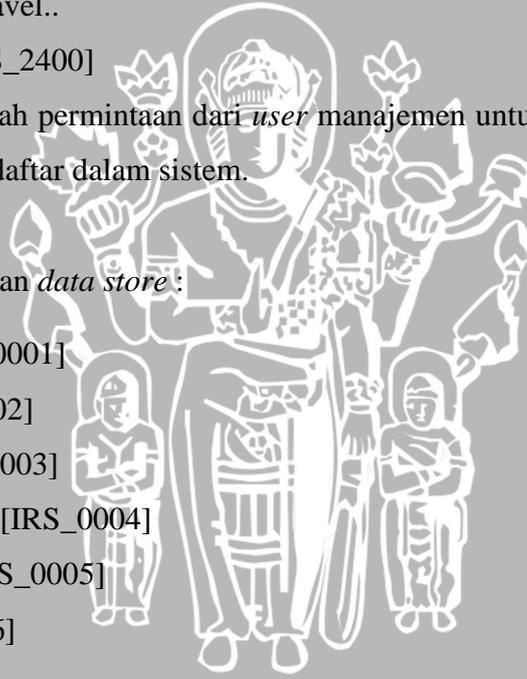
j. *tr_counter* [IRS_0003]

k. *tr_keberangkatan* [IRS_0004]

l. *tr_pemesanan* [IRS_0005]

m. *tr_tiket* [IRS_0006]

n. *user* [IRS_0007]

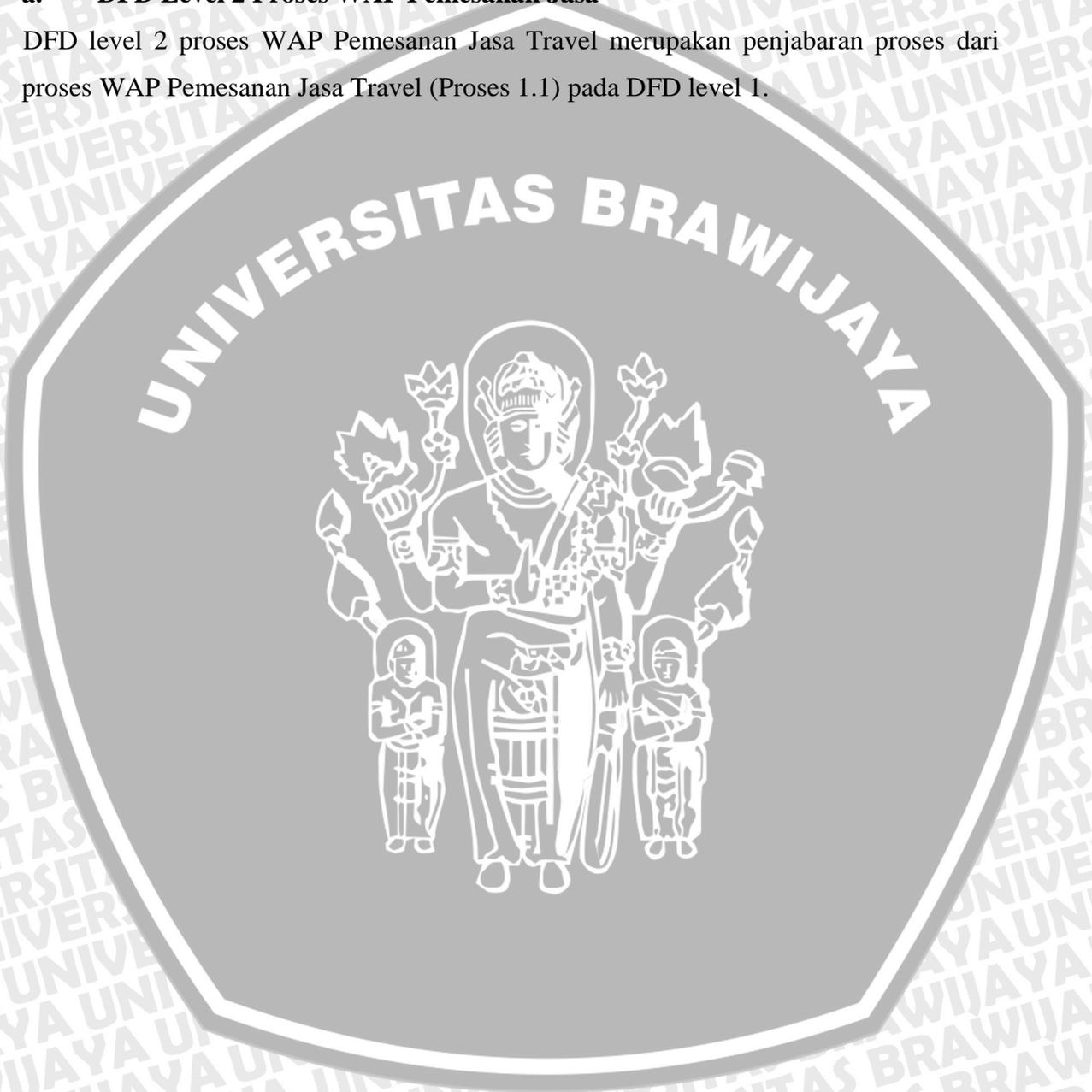


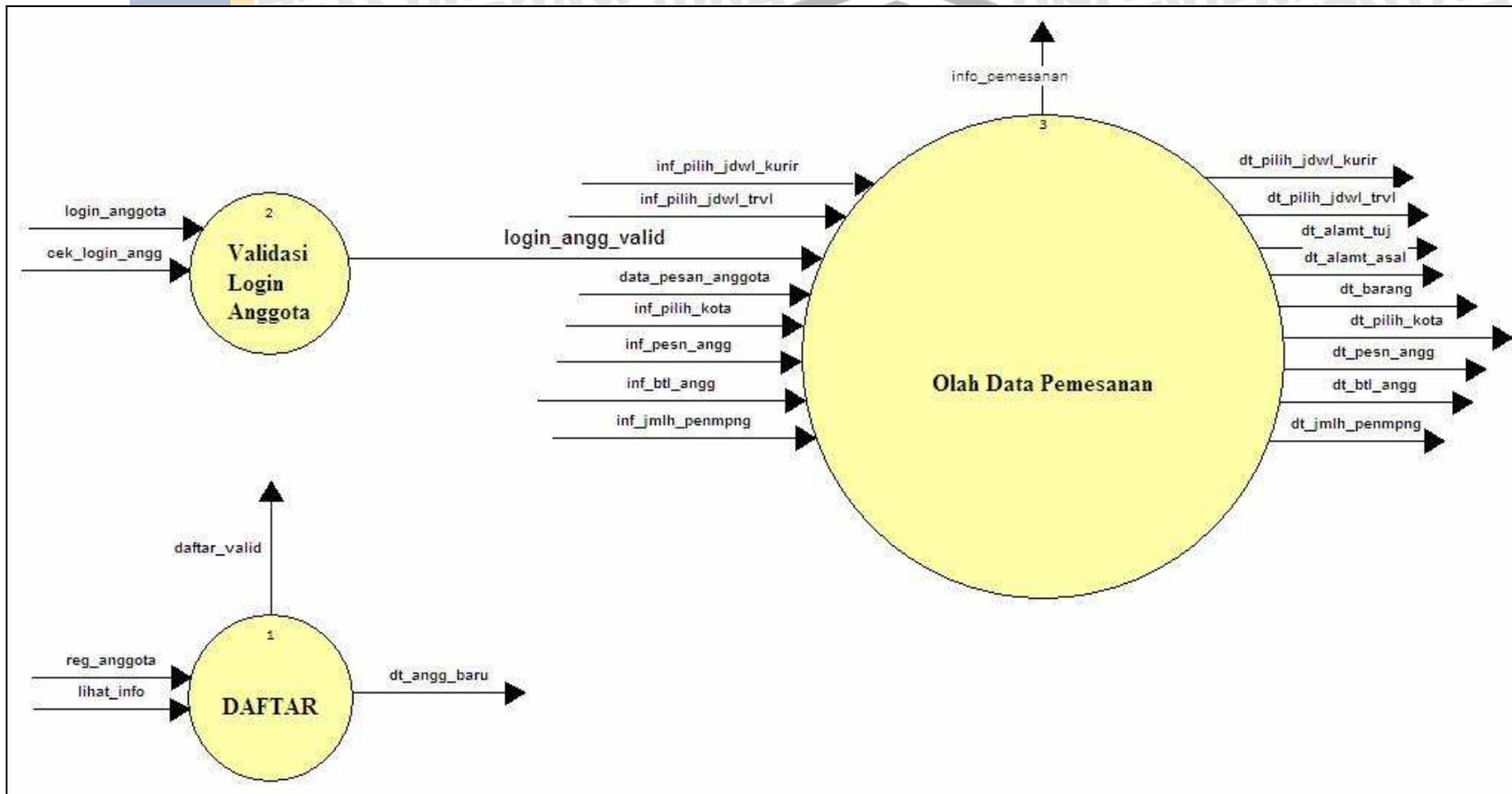
4.2.2.3 DFD Level 2

DFD Level 2 merupakan diagram yang menguraikan proses pada DFD level 1 yang mempunyai rincian proses. Pada sistem ini, mempunyai rincian proses yang akan dijelaskan berikut ini.

a. DFD Level 2 Proses WAP Pemesanan Jasa

DFD level 2 proses WAP Pemesanan Jasa Travel merupakan penjabaran proses dari proses WAP Pemesanan Jasa Travel (Proses 1.1) pada DFD level 1.





Gambar 4.7 DFD Level 2 Proses WAP Pemesanan Travel
Sumber : Analisis dan Perancangan

Gambar 4.7 merupakan DFD Level 2 proses pemesanan jasa travel dimana proses tersebut terjadi pada bagian tampilan WAP dari sistem pemesanan jasa travel ini. Sistem tersebut dibagi menjadi proses-proses yang lebih spesifik, yaitu : proses Daftar, proses Validasi Login Anggota, dan Olah Data Pemesanan. Dari gambar 4.7 maka dapat dijabarkan lebih detail proses-proses tersebut sebagai berikut:

1. Proses Daftar (1.1.1)

Proses ini digunakan untuk melakukan pendaftaran bagi pengunjung yang ingin melakukan pemesanan pemesanan travel.

Proses ini mempunyai data masukan :

- a. *reg_anggota* [IRS_0130]
- b. *lihat_info* [IRS_0170]

Proses ini mempunyai data keluaran :

- a. *daftar_valid* [IRS_0490]
- b. *dt_angg_baru* [IRS_1200]

Proses ini menggunakan *data store* :

tr_anggota [IRS_0001]

2. Proses Validasi Login Anggota (1.1.2)

Proses ini merupakan proses yang dilalui pertama kali dalam melakukan pemesanan jasa travel. Proses ini mengambil data-data yang dibutuhkan dari *data store tr_anggota*.

Proses ini mempunyai data masukan :

- a. *login_anggota* [IRS_0110]
- b. *daftar_valid* [IRS_0490]

daftar_valid adalah pengunjung yang akan memesan jasa telah melakukan registrasi dengan benar dan menjadi anggota.

- c. *cek_login_angg* [IRS_0190]

cek_login_anggota berisi data *login* yang tersimpan dalam *data store dt_anggota* untuk validasi *login* anggota.

Proses ini mempunyai data keluaran:

- a. *login_angg_valid* [IRS_4900]
login_angg_valid adalah login anggota yang telah tervalidasi setelah menerima data *cek_login_anggota*.

Proses ini menggunakan *data store*

tr_anggota [IRS_0001]

3. Proses Olah Data Pemesanan (1.1.3)

Proses ini merupakan proses pemesanan, pembatalan, dan pembayaran pemesanan jasa bagi anggota dalam tampilan WAP.

Proses ini mempunyai data masukan :

- a. *login_angg_valid* [IRS_4900]
- b. *data_pesan_anggota* [IRS_0160]
- c. *inf_bayar_angg* [IRS_0260]
- d. *inf_pilih_jdwl_kurir* [IRS_0210]
- e. *inf_pilih_jdwl_trvl* [IRS_0200]
- f. *inf_pilih_kota* [IRS_0230]
- g. *inf_pesn_angg* [IRS_0240]
- h. *inf_btl_angg* [IRS_0250]
- i. *inf_jmlh_penmpng* [IRS_0270]

Proses ini mempunyai data keluaran :

- a. *info_tarif_jadwal* [IRS_1100]
- b. *info_pemesanan* [IRS_1000]
- c. *dt_bayar* [IRS_2100]
- d. *dt_pilih_jdwl_kurir* [IRS_1400]
- e. *dt_pilih_jdwl_trvl* [IRS_1300]
- f. *dt_pilih_kota* [IRS_1500]
- g. *dt_pesn_angg* [IRS_1900]
- h. *dt_btl_angg* [IRS_2000]
- i. *dt_jmlh_penmpng* [IRS_2200]
- j. *dt_alamt_tuj* [IRS_1600]

k. *dt_alamt_asal* [IRS_1700]

l. *dt_barang* [IRS_1800]

Proses ini menggunakan *data store* :

a. *tr_anggota* [IRS_0001]

b. *tr_mobil* [IRS_0002]

c. *tr_counter* [IRS_0003]

d. *tr_keberangkatan* [IRS_0004]

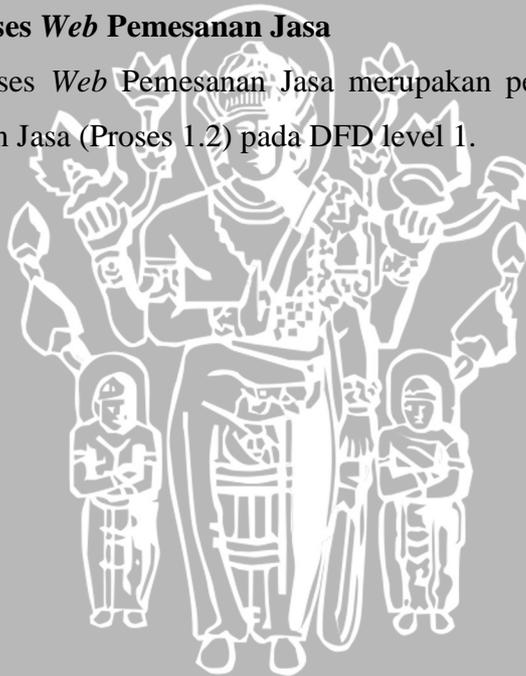
e. *tr_pemesanan* [IRS_0005]

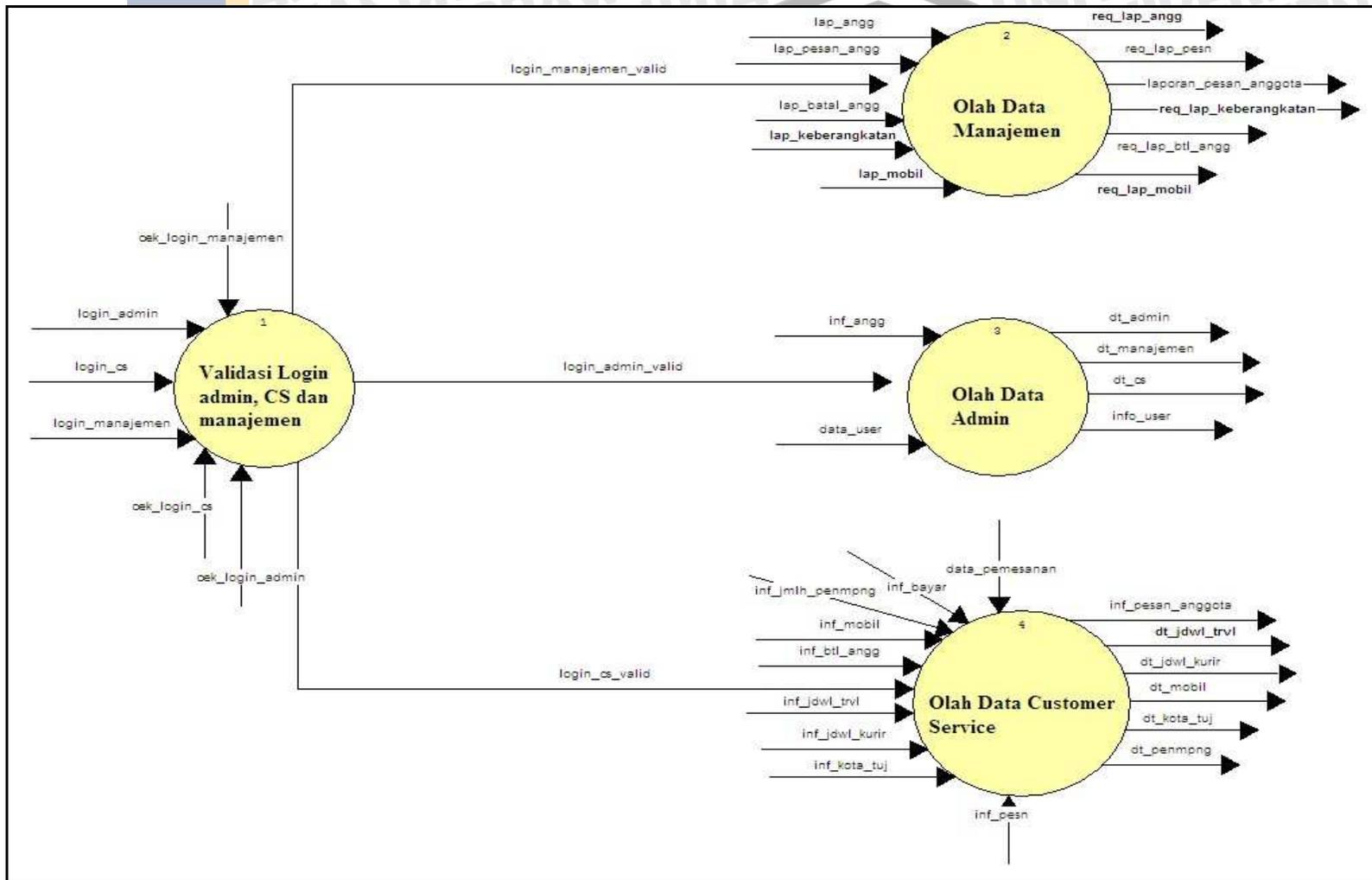
f. *tr_tiket* [IRS_0006]

g. *user* [IRS_0007]

b. DFD Level 2 Proses Web Pemesanan Jasa

DFD level 2 proses *Web Pemesanan Jasa* merupakan penjabaran proses dari proses *Web Pemesanan Jasa* (Proses 1.2) pada DFD level 1.





Gambar 4.8 DFD Level 2 Proses Web Pemesanan Travel
Sumber : Analisis dan Perancangan

Gambar 4.8 merupakan DFD Level 2 proses pemesanan jasa dimana proses tersebut terjadi pada bagian tampilan *Web* dari sistem pemesanan jasa travel ini. Sistem tersebut dibagi menjadi proses-proses yang lebih spesifik, yaitu : proses Validasi Login Admin, CS & Manajemen, Olah Data Manajemen, dan Olah Data Admin. Dari gambar 4.8, maka dapat dijabarkan lebih detail proses-proses tersebut sebagai berikut:

1. Proses Validasi Login Admin, Customer Service, & Manajemen (1.2.1)

Proses ini merupakan proses yang dilalui pertama kali oleh admin, customer service, atau manajemen dalam Sistem *Web* Pemesanan Jasa ini. Proses ini mengambil data-data yang dibutuhkan dari *data store* *dt_admin*, *dt_cs*, dan *dt_manajemen*.

Proses ini mempunyai data masukan :

- a. *login_admin* [IRS_0180]
- b. *login_manajemen* [IRS_0140]
- c. *cek_login_admin* [IRS_0280]
- d. *cek_login_manajemen* [IRS_0290]
- e. *login_cs* [IRS_0120]
- f. *cek_login_cs* [IRS_0300]

Proses ini mempunyai data keluaran:

- a. *login_admin_valid* [IRS_4200]
login_admin_valid adalah login admin yang telah tervalidasi setelah menerima data *cek_login_admin*.
- b. *login_manajemen_valid* [IRS_4300]
login_manajemen_valid adalah login manajemen yang telah tervalidasi setelah menerima data *cek_login_manajemen*.
- c. *login_cs_valid* [IRS_4400]
login_cs_valid adalah login customer service yang telah tervalidasi setelah menerima data *cek_login_cs*.

Proses ini menggunakan *data_store*:

- a. *user* [IRS_0007]

2. Proses Olah Data Manajemen (1.2.2)

Proses ini merupakan proses bagi *user* manajemen untuk melihat laporan-laporan tentang pemesanan jasa ini. Proses ini mempunyai data masukan :

- a. *login_manajemen_valid* [IRS_4300]
- b. *lap_angg* [IRS_0430]
- c. *lap_pesan_angg* [IRS_0320]
- d. *lap_batal_angg* [IRS_0330]
- e. *lap_mobil* [IRS_310]
- f. *lap_keberangkatan* [IRS_380]

Proses ini mempunyai data keluaran :

- a. *laporan_pesan_anggota* [IRS_2400]
- b. *req_lap_angg* [IRS_2100]
- c. *req_lap_pesan* [IRS_3900]
- d. *req_lap_btl_angg* [IRS_4000]
- e. *req_lap_mobil* [IRS_2500]
- f. *req_lap_keberangkatan* [IRS_1900]

Proses ini menggunakan *data store* :

- a. *tr_anggota* [IRS_0001]
- b. *tr_mobil* [IRS_0002]
- c. *tr_counter* [IRS_0003]
- d. *tr_keberangkatan* [IRS_0004]
- e. *tr_pemesanan* [IRS_0005]
- f. *tr_tiket* [IRS_0006]
- g. *user* [IRS_0007]

3. Proses Olah Data Customer Service (1.2.3)

Proses ini merupakan proses pengolahan data oleh *user* cs.

Proses ini mempunyai data masukan :

- a. *login_cs_valid* [IRS_4400]
- b. *data_jadwal* [IRS_0150]
- c. *inf_jdwl_trvl* [IRS_0320]
- d. *inf_jdwl_kurir* [IRS_0330]

- e. *inf_mobil* [IRS_0340]
- f. *inf_kota_tuj* [IRS_0350]
- g. *inf_pesn* [IRS_0360]
- h. *inf_btl_angg* [IRS_0370]

Proses ini mempunyai data keluaran :

- a. *info_pesan_anggota* [IRS_2300]
- b. *dt_jdwl_trvl* [IRS_2800]
- c. *dt_jdwl_kurir* [IRS_0330]
- d. *dt_mobil* [IRS_3100]
- e. *dt_kota_tuj* [IRS_3200]
- f. *dt_jmlh_penmpng* [IRS_3000]

Proses ini menggunakan *data store* :

- a. *tr_anggota* [IRS_0001]
- b. *tr_mobil* [IRS_0002]
- c. *tr_counter* [IRS_0003]
- d. *tr_keberangkatan* [IRS_0004]
- e. *tr_pemesanan* [IRS_0005]
- f. *tr_tiket* [IRS_0006]
- g. *user* [IRS_0007]

4. Proses Olah Data Admin (1.2.4)

Proses ini merupakan proses pengolahan data oleh *user* admin perangkat lunak. Data masukan dan keluaran dari proses ini berasal dari *data store anggota*, *data store admin*, dan *manajemen*.

Proses ini mempunyai data masukan :

- a. *login_adm_valid* [IRS_4200]
- b. *data_user* [IRS_0500]
- c. *inf_angg* [IRS_0310]

Proses ini mempunyai data keluaran :

- a. *dt_cs* [IRS_2700]
- b. *info_user* [IRS_2500]

- c. *dt_admin* [IRS_2800]
- d. *dt_manajemen* [IRS_2600]

Proses ini menggunakan *data store* :

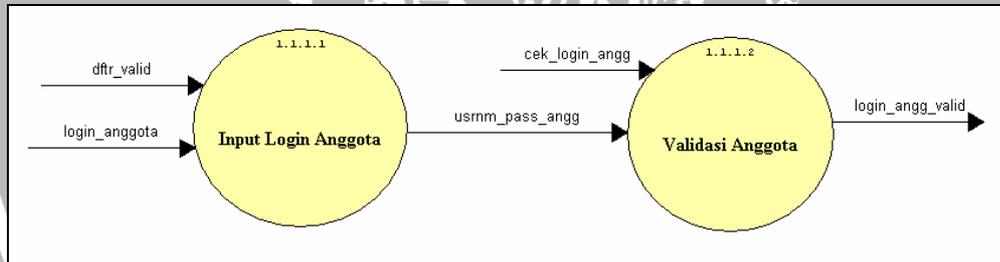
- a. *tr_anggota* [IRS_0001]
- b. *user* [IRS_0007]

4.2.2.4 DFD Level 3

DFD Level 3 merupakan diagram yang menguraikan proses pada DFD level 2 yang mempunyai rincian proses. Pada sistem ini, mempunyai rincian proses yang akan dijelaskan pada subbab berikut ini.

a. DFD Level 3 Validasi Login Anggota

DFD Level 3 Proses Validasi Login Anggota merupakan penjabaran dari DFD level 2 Proses WAP Pemesanan Pemesanan travel pada proses Validasi Login Anggota (proses 1.1.1). Penjabaran tersebut dapat dijelaskan dalam gambar 4.9.



Gambar 4.9 DFD Level 3 Proses Validasi Login Anggota
 Sumber : *Analisis dan Perancangan*

Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Input Login Anggota (1.1.1.1)

Merupakan proses untuk memasukkan data login dari *user* yang berupa *username* dan *password*.

Proses ini mempunyai data masukan:

- a. *login_anggota* [IRS_0110]
- b. *daftar_valid* [IRS_0490]

Proses ini mempunyai data keluaran:

usnm_pass_angg [IRS_4500]



2. Proses Validasi Anggota (1.1.1.2)

Proses ini mengolah berupa validasi data login yaitu proses Input Login Anggota yang dicocokkan dengan data pada *dt_anggota*. Apabila data valid maka *user* anggota dapat melakukan proses berikutnya, dan apabila data tidak valid maka akan didapatkan pesan kesalahan dan diberikan pilihan untuk input *login* kembali.

Proses ini mempunyai data masukan :

- a. *usrnm_pass_angg* [IRS_4500]
- b. *cek_login_angg* [IRS_0190]

Proses ini mempunyai data keluaran :

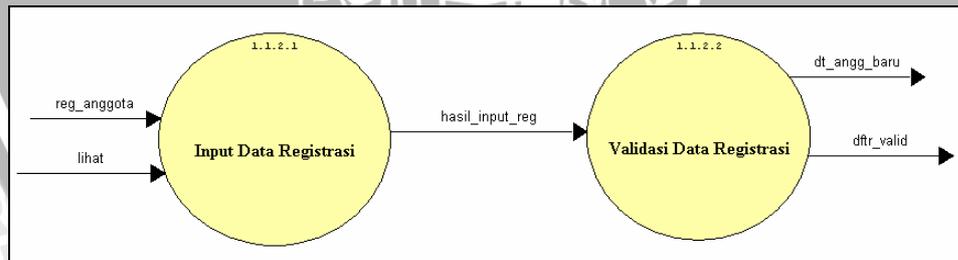
login_angg_valid [IRS_4100]

Proses ini menggunakan *data store*:

tr_anggota [IRS_0001]

b. **DFD Level 3 Daftar**

Gambar 4.10 merupakan DFD Level 3 proses Daftar dimana proses tersebut dapat dibagi menjadi proses yang lebih spesifik lagi, yaitu : proses Input Data Registrasi, dan Validasi Data Registrasi.



Gambar 4.10 DFD Level 3 Proses Daftar
 Sumber : Analisis dan Perancangan

Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Input Data Registrasi (1.1.2.1)

Merupakan proses bagi pengunjung yang ingin melakukan pemesanan perjalanan dengan mendaftarkan diri menjadi anggota.

Proses ini mempunyai data masukan:

- a. *reg_anggota* [IRS_0130]



b. *lihat_info* [IRS_0170]

Proses ini mempunyai data keluaran:

hasil_input_reg [IRS_4600]

2. Proses Validasi Data Registrasi (1.1.2.2)

Proses ini merupakan validasi *hasil_input_reg* yang merupakan keluaran proses sebelumnya. Bila *hasil_input_reg* adalah valid maka *user* dapat melakukan proses selanjutnya. Bila tidak valid maka *user* pengunjung diminta memasukkan *reg_anggota* sekali lagi.

Proses ini mempunyai data masukan :

hasil_input_reg [IRS_4600]

Proses ini mempunyai data keluaran :

a. *daftar_valid* [IRS_0490]

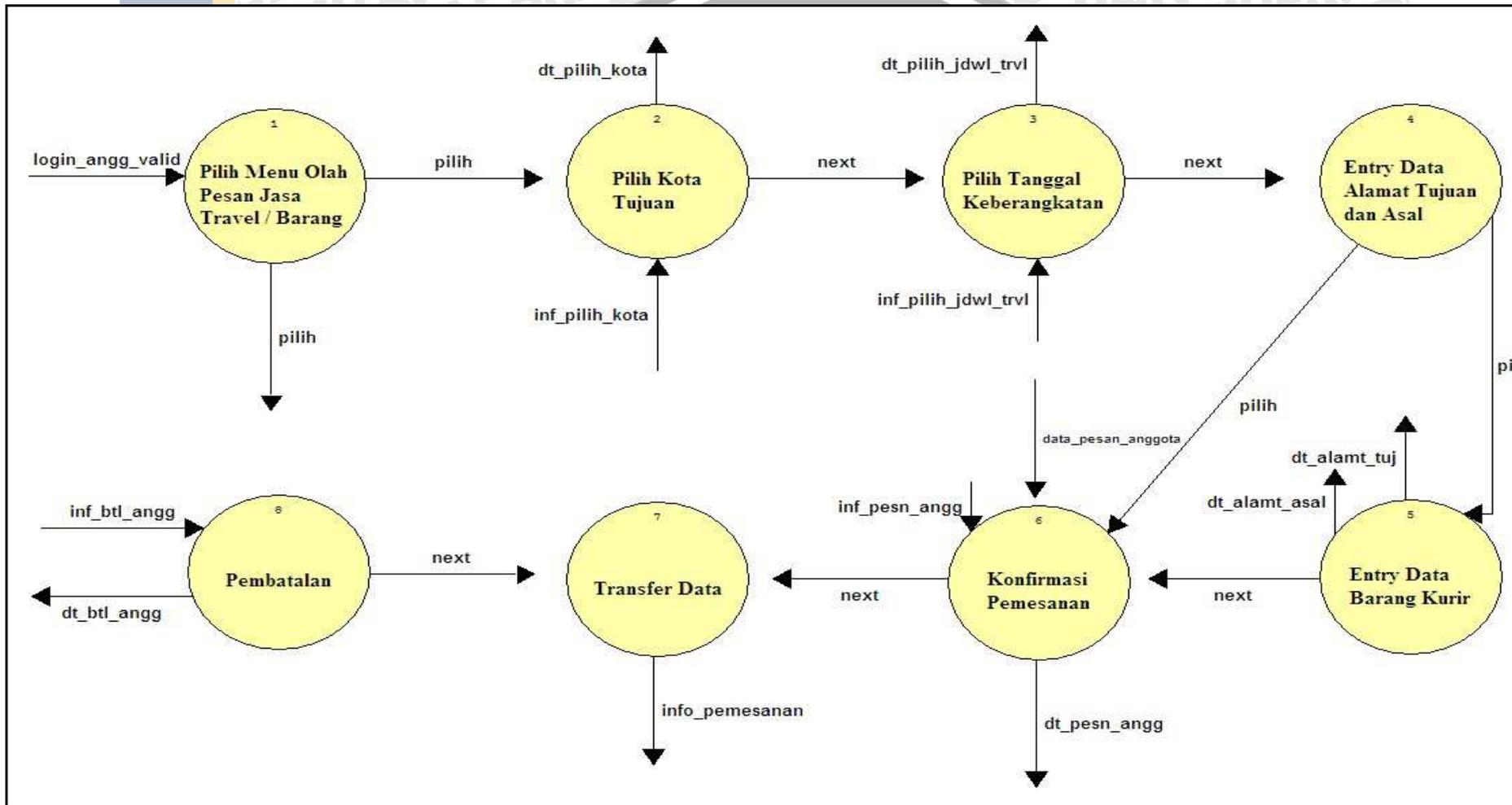
b. *dt_angg_baru* [IRS_1200]

Proses ini menggunakan *data store*:

tr_anggota [IRS_0001]

c. **DFD Level 3 Olah Data Pemesanan**

Gambar 4.11 merupakan DFD Level 3 proses Olah Data Pemesanan yang merupakan penjabaran dari DFD level 2 Proses WAP Pemesanan Jasa pada proses Olah Data Pemesanan (proses 1.1.3). Penjabaran tersebut dapat dijelaskan dalam gambar 4.11.



Gambar 4.11 DFD Level 3 Olah data Pemesanan

Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Pilih Menu Olah Pesan (1.1.3.1)

Proses ini merupakan proses pilihan menu yang diinginkan bagi *user cs* untuk menuju pada proses berikutnya.

Proses ini mempunyai data masukan:

login_angg_valid [IRS_4100]

Proses ini mempunyai data keluaran:

pilih [IRS_4700]

2. Pilih Kota Tujuan (1.1.3.2)

Proses ini merupakan proses memilih kota tujuan keberangkatan travel oleh anggota.

Proses ini mempunyai data masukan :

- a. *pilih* [IRS_4700]
- b. *inf_pilih_kotal* [IRS_0230]

Proses ini mempunyai data keluaran :

- a. *dt_pilih_kota* [IRS_1500]
- b. *next* [IRS_4800]

Proses ini menggunakan *data store*:

tr_keberangkatan [IRS_0004]

3. Pilih Tanggal Keberangkatan (1.1.3.3)

Proses ini merupakan proses memilih tanggal keberangkatan travel oleh anggota.

Proses ini mempunyai data masukan :

- a. *next* [IRS_4800]
- b. *inf_pilih_jdwl_trvl* [IRS_0200]

c. *inf_pilih_jdwl_kurir* [IRS_0210]

Proses ini mempunyai data keluaran :

- a. *dt_pilih_jdwl_trvl* [IRS_1300]
- b. *dt_pilih_jdwl_kurir* [IRS_1400]
- c. *next* [IRS_4800]

Proses ini menggunakan *data store*:

- a. *tr_keberangkatan* [IRS_0004]

4. Entry Data Alamat Asal dan Alamat Tujuan (1.1.3.4)

Proses ini merupakan proses entry data alamat tujuan dan alamat penjemputan yang diisi oleh anggota.

Proses ini mempunyai data masukan :

- a. *next* [IRS_4800]
- b. *inf_alamt_asal* [IRS_0420]
- c. *inf_alamt_tuj* [IRS_0410]

Proses ini mempunyai data keluaran :

- a. *dt_alamt_asal* [IRS_1700]
- b. *dt_alamt_tuj* [IRS_1600]
- b. *next* [IRS_4800]

5. Konfirmasi Pemesanan (1.1.3.5)

Proses ini merupakan proses konfirmasi pemesanan jasa travel oleh anggota.

Proses ini mempunyai data masukan :

- a. *next* [IRS_4800]
- b. *data_pesan_anggota* [IRS_0160]
- c. *inf_psn_angg* [IRS_0240]

Proses ini mempunyai data keluaran :

- a. *next* [IRS_4800]
- b. *dt_pesn_angg* [IRS_1900]
- c. *info_tarif_jdwl* [IRS_1100]

Proses ini menggunakan *data store*:

- a. *tr_pemesanan* [IRS_0005]

6. Pembatalan (1.1.3.6)

Proses ini merupakan proses pembatalan pemesanan jasa oleh anggota.

Proses ini mempunyai data masukan :

- a. *pilih* [IRS_4700]
- b. *inf_btl_angg* [IRS_0250]

Proses ini mempunyai data keluaran :

- a. *dt_btl_angg* [IRS_2000]
- b. *next* [IRS_4800]

Proses ini menggunakan *data store*:

- tr_pemesanan* [IRS_0005]

7. Transfer Data (1.1.3.7)

Proses ini merupakan proses validasi pemesanan.

Proses ini mempunyai data masukan :

- a. *next* [IRS_4800]

Proses ini mempunyai data keluaran :

- a. *info_pemesanan* [IRS_1000]

Proses ini menggunakan *data store*:



tr_pemesanan [IRS_0005]

8. Entry Data Barang Kurir (1.1.3.8)

Proses ini merupakan proses pemasukan keterangan barang yang akan dikirim oleh anggota.

Proses ini mempunyai data masukan :

- a. *next* [IRS_4800]
- b. *inf_barang* [IRS_0220]

Proses ini mempunyai data keluaran :

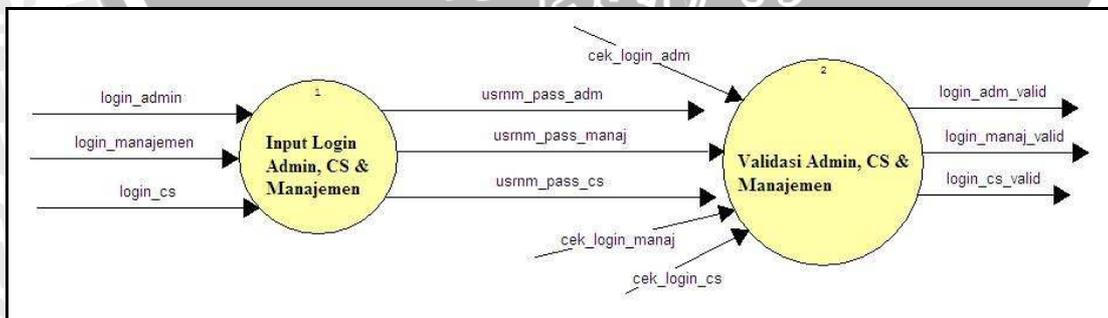
- a. *dt_barang* [IRS_1800]
- b. *next* [IRS_4800]

Proses ini menggunakan *data store*:

tr_pemesanan [IRS_0005]

d. **DFD Level 3 Validasi Login Admin, Admin PL, & Manajemen**

Gambar 4.12 merupakan DFD Level 3 proses Validasi Login Customer Service, Admin, & Manajemen yang merupakan penjabaran dari DFD level 2 Proses *Web Pemesanan Jasa Travel* pada proses Validasi Login Customer Service, Admin, & Manajemen (proses 1.2.1). Penjabaran tersebut dapat dijelaskan dalam gambar 4.12



Gambar 4.12 DFD Level 3 Proses Validasi Login Admin & Manajemen
 Sumber : *Analisis dan Perancangan*



Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Input Login Admin, Customer Service, & Manajemen (1.2.1.1)

Merupakan proses untuk memasukkan data login dari *user* admin, customer service, dan manajemen yang berupa *username* dan *password*.

Proses ini mempunyai data masukan:

- a. *login_admin* [IRS_0180]
- b. *login_manajemen* [IRS_0140]
- c. *login_cs* [IRS_0120]

Proses ini mempunyai data keluaran:

- a. *usrnm_pass_adm* [IRS_5000]
- b. *usrnm_pass_manaj* [IRS_5100]
- c. *usrnm_pass_cs* [IRS_5200]

2. Validasi Admin, Customer Service, & Manajemen (1.2.1.2)

Proses ini mengolah berupa validasi data login proses Input Login Admin, Customer Service, & Manajemen yang dicocokkan dengan data pada *dt_admin*, *dt_cs*, dan *dt_manajemen*. Data valid maka *user* dapat melakukan proses berikutnya, dan apabila data tidak valid maka *user* akan mendapatkan pesan kesalahan dan diberikan pilihan untuk input *login* kembali.

Proses ini mempunyai data masukan :

- a. *usrnm_pass_adm* [IRS_4900]
- b. *usrnm_pass_manaj* [IRS_5000]
- c. *usrnm_pass_cs* [IRS_5100]
- d. *cek_login_admin* [IRS_0280]
- e. *cek_login_manajemen* [IRS_0290]
- f. *cek_login_cs* [IRS_0300]

Proses ini mempunyai data keluaran :

- a. *login_admin_valid* [IRS_4200]

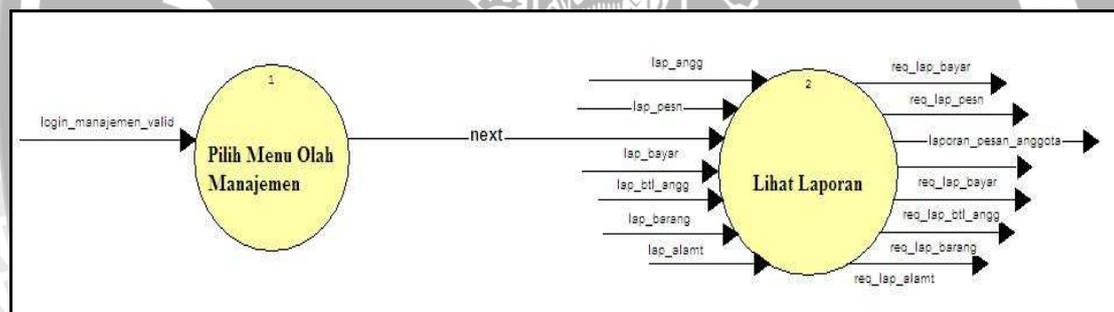
- b. *login_manaj_valid* [IRS_4300]
- c. *login_adm_pl_valid* [IRS_4400]

Proses ini menggunakan *data store*:

- a. *user* [IRS_0007]

e. DFD Level 3 Olah Data Manajemen

Gambar 4.13 merupakan DFD Level 3 proses Olah Data Manajemen yang merupakan penjabaran dari DFD level 2 Proses *Web* Pemesanan Jasa Travel pada proses Olah Data Manajemen (proses 1.2.2). Penjabaran tersebut dapat dijelaskan dalam gambar 4.13.



Gambar 4.13 DFD Level 3 Proses Olah Data Manajemen

Sumber : *Analisis dan Perancangan*

Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Masuk Menu Olah Manajemen (1.2.2.1)

Merupakan proses menampilkan menu untuk melihat laporan bagi *user* manajemen.

Proses ini mempunyai data masukan:

login_manaj_valid [IRS_4300]

Proses ini mempunyai data keluaran:

next [IRS_4800]

2. Lihat Laporan (1.2.2.2)



Proses ini adalah proses melihat laporan bagi *user* manajemen.

Proses ini mempunyai data masukan :

- a. *next* [IRS_4800]
- b. *lap_angg* [IRS_0430]
- c. *lap_barang* [IRS_0480]
- d. *lap_pesn* [IRS_0440]
- e. *lap_btl_angg* [IRS_0450]
- f. *lap_bayar* [IRS_0460]
- g. *lap_alamt* [IRS_0470]

Proses ini mempunyai data keluaran :

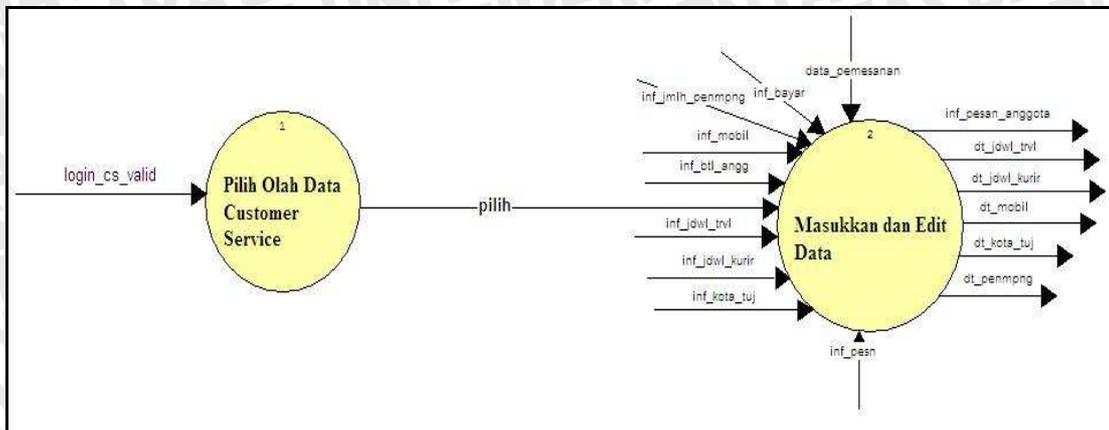
- a. *laporan_pesan_anggota* [IRS_2400]
- b. *req_lap_barang* [IRS_3600]
- c. *req_lap_alamt* [IRS_3800]
- d. *req_lap_pesn* [IRS_3900]
- e. *req_lap_btl_angg* [IRS_4000]
- f. *req_lap_bayar* [IRS_3300]

Proses ini menggunakan *data store*:

- a. *tr_anggota* [IRS_0001]
- b. *tr_mobil* [IRS_0002]
- c. *tr_counter* [IRS_0003]
- d. *tr_keberangkatan* [IRS_0004]
- e. *tr_pemesanan* [IRS_0005]
- f. *tr_tiket* [IRS_0006]

f. DFD Level 3 Olah Data Admin

Gambar 4.14 merupakan DFD Level 3 proses Olah Data Admin yang merupakan penjabaran dari DFD level 2 Proses *Web* Pemesanan Tiket pada proses Olah Data Admin (proses 1.2.3). Penjabaran tersebut dapat dijelaskan dalam gambar 4.14.



Gambar 4.14 DFD Level 3 Proses Olah Data Admin
Sumber : *Analisis dan Perancangan*

Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Pilih Menu Olah Customer Service (1.2.3.1)

Merupakan proses menampilkan menu untuk bagi *user cs*.

Proses ini mempunyai data masukan:

login_cs_valid [IRS_4400]

Proses ini mempunyai data keluaran:

pilih [IRS_4700]

2. Masukkan & Edit Data (1.2.3.2)

Proses ini adalah proses memasukkan atau mengedit data bagi *user cs* ke *data store*.

Proses ini mempunyai data masukan :

- a. *pilih* [IRS_4700]
- b. *data_pemesanan* [IRS_0150]
- c. *inf_jdwl_trvl* [IRS_0320]
- d. *inf_jdwl_kurir* [IRS_0330]
- e. *inf_mobil* [IRS_0340]
- f. *inf_kota_tuj* [IRS_0350]
- g. *inf_pesn* [IRS_0360]

- h. *inf_btl_angg* [IRS_0370]
- i. *inf_bayar* [IRS_0380]
- j. *inf_jmlh_penmpng* [IRS_0390]

Proses ini mempunyai data keluaran :

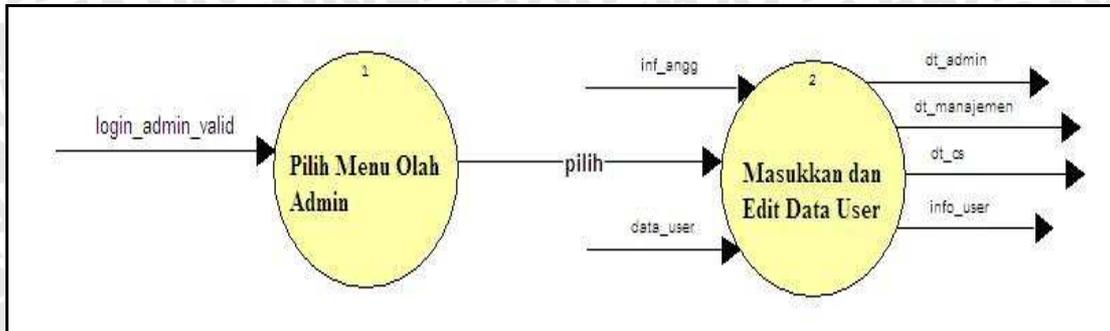
- a. *info_pesan_anggota* [IRS_2300]
- b. *dt_jdwl_trvl* [IRS_2800]
- c. *dt_jdwl_kurir* [IRS_2900]
- d. *dt_mobil* [IRS_3100]
- e. *dt_kota_tuj* [IRS_3200]
- f. *dt_jmlh_penmpng* [IRS_3000]

Proses ini menggunakan *data store*:

- a. *tr_anggota* [IRS_0001]
- b. *tr_mobil* [IRS_0002]
- c. *tr_counter* [IRS_0003]
- d. *tr_keberangkatan* [IRS_0004]
- e. *tr_pemesanan* [IRS_0005]
- f. *tr_tiket* [IRS_0006]

g. DFD Level 3 Olah Data Admin Perangkat Lunak

Gambar 4.15 merupakan DFD Level 3 proses Olah Data Admin Perangkat Lunak yang merupakan penjabaran dari DFD level 2 Proses *Web Pemesanan Jasa Travel* pada proses Olah Data Admin (proses 1.2.4). Penjabaran tersebut dapat dijelaskan dalam gambar 4.15.



Gambar 4.15 DFD Level 3 Proses Olah Data Admin
 Sumber : Analisis dan Perancangan

Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Pilih Menu Olah Admin PL(1.2.4.1)

Merupakan proses menampilkan menu untuk bagi *user* admin.

Proses ini mempunyai data masukan:

login_admin_valid [IRS_4200]

Proses ini mempunyai data keluaran:

pilih [IRS_4700]

2. Masukkan & Edit Data User(1.2.4.2)

Proses ini adalah proses memasukkan atau mengedit data *user* ke *data store*.

Proses ini mempunyai data masukan :

- a. *pilih* [IRS_4700]
- b. *inf_angg* [IRS_0310]
- c. *data_user* [IRS_0500]

Proses ini mempunyai data keluaran :

- a. *info_user* [IRS_2500]
- b. *dt_admin* [IRS_2800]
- c. *dt_manaj* [IRS_2600]
- d. *dt_cs* [IRS_2700]



Proses ini menggunakan *data store*:

- a. *user* [IRS_0007]

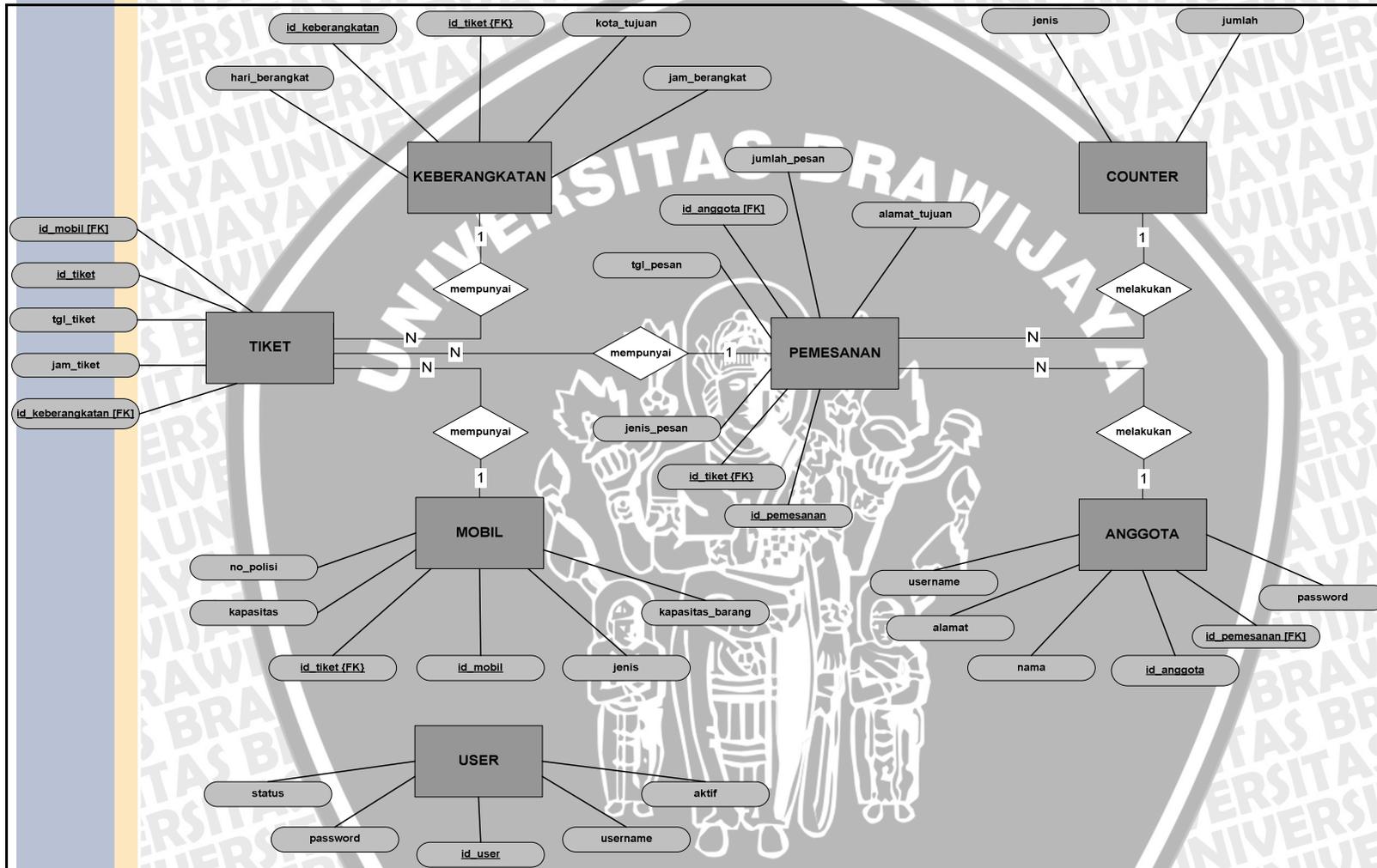
4.2.3 Perancangan Basis Data

Perancangan basis data dilakukan agar Sistem Pemesanan Jasa Travel Berbasis WAP ini dapat menjadi basis data yang dapat digunakan sebagaimana mestinya yaitu sebagai ruang penyimpanan, dapat diakses dengan cepat dan mudah untuk dimanipulasi pada pengembangan perangkat lunak dalam waktu yang akan datang. Perancangan basis Pembuatan ERD (*Entity Relational Diagram*)

4.2.3.1 ERD (*Entity Relational Diagram*)

Diagram ER digunakan untuk menggambarkan hubungan entitas satu dengan lainnya dengan memperlihatkan hubungan antar atribut yang akan dijadikan *key* untuk berelasi antar tabel.

Entity relationship diagram (ERD) digunakan untuk memodelkan struktur data dan hubungan antar data. ERD menggambarkan keadaan dunia nyata (*real world*) yang diwakili dalam bentuk entiti, atribut serta relasi. ERD memperlihatkan hubungan antara atribut yang akan dijadikan kunci untuk berelasi antar tabel. Entitas tiket, keberangkatan, pemesanan, anggota, counter, mobil dan user masing-masing memiliki struktur tabel basis data. ERD dari basis data Sistem Pemesanan Jasa Travel menggunakan layanan WAP ditunjukkan dalam Gambar 4.16.

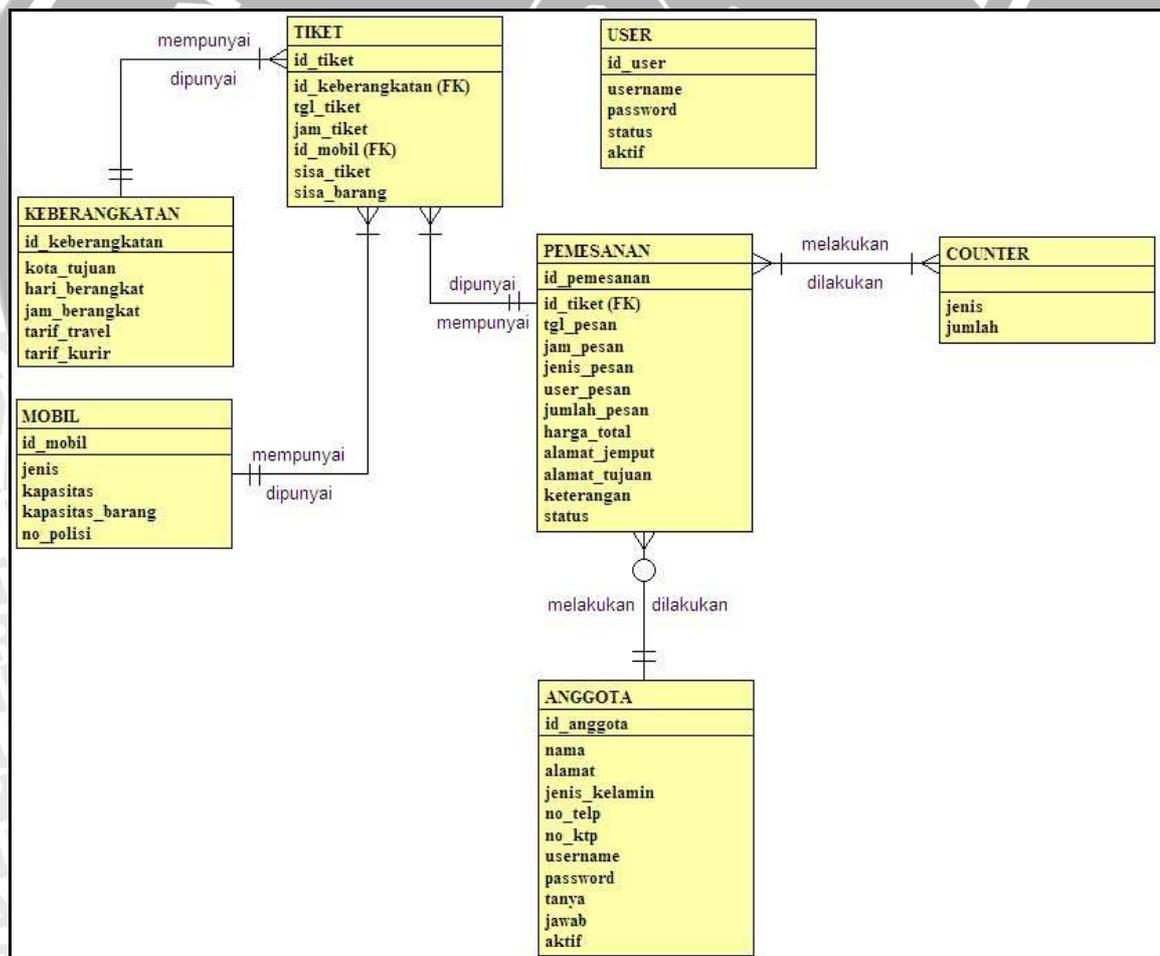


Gambar 4.16 ER diagram

Sumber: Perancangan

4.2.3.2 Normalisasi Data

Pada Sistem Pemesanan dan Pengiriman Barang menggunakan Travel dengan Layanan WAP, normalisasi data dibutuhkan untuk mengurangi pengulangan (redundansi) data di setiap tabel dan mempercepat proses pencarian data di dalam basis data. Normalisasi sering dilakukan sebagai suatu uji coba pada suatu relasi secara berkelanjutan untuk menentukan apakah relasi tersebut sudah baik atau masih melanggar aturan-aturan standar yang diberlakukan pada suatu relasi yang normal (sudah dapat dilakukan proses *insert*, *update*, *delete*, dan *modify* pada satu atau beberapa atribut tanpa mempengaruhi integritas data dalam relasi tersebut). Tabel-tabel yang akan digunakan untuk menyimpan data dinormalisasi menjadi tujuh buah entitas (tabel) yaitu tiket, keberangkatan, pemesanan, anggota, counter, mobil dan user. Rancangan normalisasi tabel-tabel tersebut diperlihatkan dalam Gambar 4.17.



Gambar 4.17 Normalisasi Tabel dalam Basis Data Sistem Pemesanan Travel

4.2.3.3 Data Object Description

Data object description menjelaskan secara rinci mengenai atribut-atribut yang dimiliki oleh masing-masing tabel yang ada dalam basis data sesuai dengan *entity relationship diagram*. *Data object description* dapat dipisahkan menjadi dua bagian, yaitu:

1. *Data type definition* menjelaskan tipe data yang digunakan oleh atribut pada masing-masing tabel yang ada dalam basis data Sistem Pemesanan Jasa Travel.
2. *Data field description* menjelaskan keterangan setiap atribut pada masing-masing tabel yang ada dalam basis data Sistem Pemesanan Jasa Travel.

4.2.3.3.1 Data Type Definition

Data type definition menjelaskan mengenai tipe data yang digunakan setiap tabel basis data Sistem Pemesanan Jasa Travel. Tabel-tabel yang terdapat dalam basis data tersebut ditunjukkan dalam Tabel 4.3 sampai dengan Tabel 4.9.

Tabel 4.3 Entitas dan Atribut tr_ anggota

Field	Tipe	Lebar	Keterangan
id_anggota	VARCHAR	5	PRYMARY KEY, AUTO INCREMENT
nama	VARCHAR	100	
alamat	VARCHAR	100	
jenis_kelamin	enum (pria;wanita)		
no_telp	VARCHAR	20	
no_ktp	VARCHAR	30	
username	VARCHAR	20	
password	VARCHAR	20	
tanya	VARCHAR	100	
jawab	VARCHAR	100	
aktif	ENUM (1;0)		

Entitas tr_ anggota merupakan Tabel anggota yang digunakan untuk menyimpan data-data tentang anggota.

Tabel 4.4 Entitas dan Atribut Counter

Field	Tipe	Lebar	Keterangan
jenis	VARCHAR	10	PRYMARY KEY, NOT NULL
jumlah	INT	5	NOT NULL

Entitas counter merupakan Tabel counter yang digunakan untuk menyimpan data jenis pesanan.

Tabel 4.5 Entitas dan Atribut Keberangkatan

Field	Tipe	Lebar	Keterangan
id_keberangkatan	INT	10	PRYMARY KEY
kota_tujuan	VARCHAR	40	
hari_berangkat	VARCHAR	50	
jam_berangkat	VARCHAR	50	
tarif_travel	INT	8	
tarif_kurir	INT	8	

Entitas keberangkatan merupakan Tabel keberangkatan yang digunakan untuk menyimpan data jadwal dan tarif pemesanan jasa travel.

Tabel 4.6 Entitas dan Atribut Mobil

Field	Tipe	Lebar	Keterangan
id_mobil	INT	5	PRIMARY KEY
jenis	VARCHAR	50	FOREIGN KEY
kapasitas	INT	3	
kapasitas_barang	INT	2	
no_polisi	VARCHAR	50	

Entitas Mobil merupakan Tabel mobil yang digunakan untuk menyimpan data armada mobil yang digunakan.

Tabel 4.7 Entitas dan Atribut Pemesanan

Field	Tipe	Lebar	Keterangan
id_pemesanan	VARCHAR	5	PRIMARY KEY
id_tiket	INT	5	FOREIGN KEY
tgl_pesan	DATE		NOT NULL
jam_pesan	VARCHAR	8	NOT NULL
jenis_pesan	ENUM (travel;kurir)		NOT NULL
user_pesan	VARCHAR	20	NOT NULL
jumlah_pesan	INT	3	NOT NULL
harga_total	VARCHAR	8	NOT NULL

alamat_jemput	VARCHAR	200	NOT NULL
alamat_tujuan	VARCHAR	200	NOT NULL
keterangan	VARCHAR	200	NOT NULL
status	ENUM (0;1)		NOT NULL

Entitas Pemesanan merupakan tabel pemesanan yang berisi informasi pemesanan jasa yang dilakukan oleh anggota. Entitas pemesanan merupakan tabel tr_pemesanan yang digunakan untuk menyimpan data pemesanan anggota.

Tabel 4.8 Entitas dan Atribut Tiket

Field	Type	Lebar	Keterangan
id_tiket	INT	5	PRYMARY KEY
id_keberangkatan	INT	5	FOREIGN KEY
tanggal_tiket	DATE		
jam_tiket	VARCHAR	5	
id_mobil	INT	5	FOREIGN KEY
sisa_tiket	INT	3	
sisa_barang	INT	2	

Entitas tiket merupakan Tabel tiket yang digunakan untuk menampilkan sisa tiket yang masih bisa dipesan oleh anggota.

Tabel 4.9 Entitas dan Atribut User

Field	Type	Lebar	Keterangan
id_user	INT	3	PRYMARY KEY
username	VARCHAR	20	
password	VARCHAR	20	
status	ENUM (admin:mj;cs)		
aktif	ENUM (1;0)		

Entitas User merupakan Tabel user yang digunakan untuk menyimpan data user yang terdaftar dalam sistem pemesanan jasa travel.

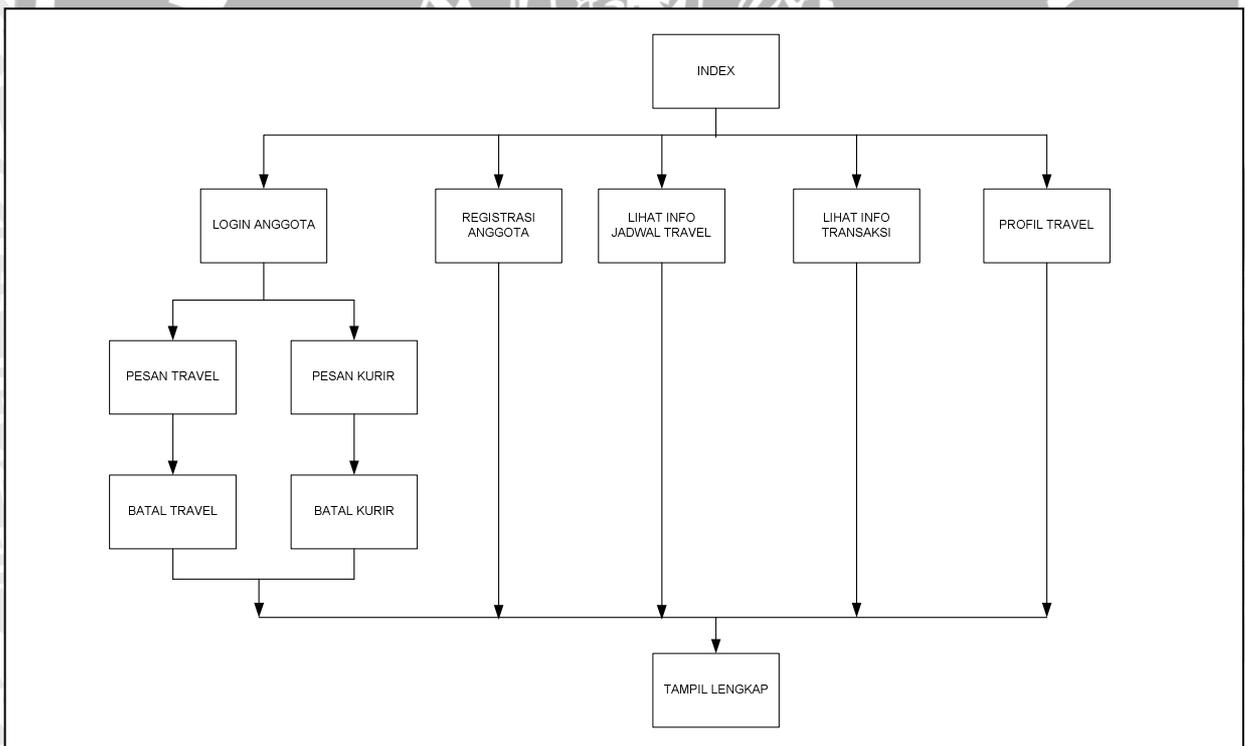
4.2.4 Site Map Sistem

Site Map merupakan penggambaran dari menu-menu yang diterapkan pada sistem yang akan dibuat. Dari menu-menu yang akan dibuat maka dapat diketahui alur pengaksesan dari tiap halaman.

Situs WAP Pemesanan Jasa Travel menggunakan halaman *index* yang akan diproses pada *server*. Halaman ini berisi halaman selamat datang yang akan ditampilkan selama lima detik, dan dilanjutkan dengan halaman utama yang berisi lima *link* yang dapat dipilih menuju ke halaman-halaman yang berbeda. *User* anggota yang telah melakukan *login* dan tervalidasi dapat mengakses keseluruhan sistem sesuai batasan yang telah diberikan oleh sistem.

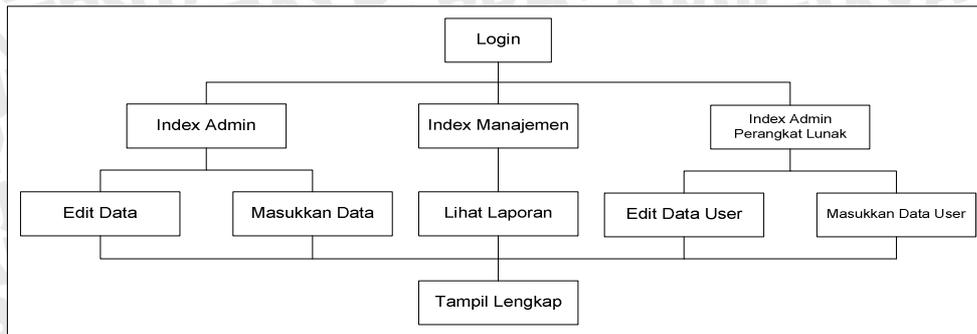
Antarmuka *web* digunakan oleh *user* admin dan manajemen untuk memasukkan atau mengubah data, dan melihat laporan-laporan yang diberikan oleh sistem. Sedangkan antarmuka WAP digunakan oleh *user* anggota dan pengunjung.

Site map sistem yang dibuat dapat ditunjukkan pada gambar berikut:



Gambar 4.18 Site Map Sistem Pemesanan Jasa Travel (Antarmuka WAP)

Sumber : Analisis dan Perancangan



Gambar 4.19 Site Map Sistem Pemesanan Tiket Bus Malam (Antarmuka Web)
 Sumber : Analisis dan Perancangan

4.2.5 Perancangan Antarmuka User

Antarmuka merupakan bagian yang menghubungkan antara sistem dengan *user*. Dengan demikian suatu perancangan antarmuka yang baik akan memudahkan *user* untuk berinteraksi dengan sistem yang dibuat.

4.2.5.1 Antarmuka User Pengunjung dan Anggota

User pengunjung merupakan *user* yang mempunyai hak akses paling terbatas karena untuk mengakses sistem ini, *user* pengunjung tidak memerlukan *login*.

a. Tampilan Index Pengunjung dan Anggota

Menu ini merupakan halaman yang pertama kali diakses *user* pada saat alamat URL situs ini diakses melalui telepon genggam. Pada halaman ini terdapat lima *link*, yaitu: Pesan Travel, Registrasi Anggota, Lihat Informasi Transaksi, Lihat Info Jadwal Travel, Lihat Info Jadwal Kurir dan Profil Jasa Travel. *Link* yang dapat diakses oleh *user* pengunjung adalah Registrasi Anggota Baru, Lihat Info Jadwal Travel, Lihat Info Jadwal Kurir dan Profil Jasa Travel. Sebelum memasuki halaman ini akan muncul halaman *index* yang menampilkan ucapan *selamat datang* selama lima detik.

BANNER	
O	PESAN TRAVEL
O	REGISTRASI ANGGOTA
O	LIHAT JADWAL TRAVEL
O	LIHAT JADWAL KURIR
O	LIHAT INFORMASI TRANSAKSI
O	PROFIL JASA TRAVEL

FOOTER	
--------	--

Gambar 4.20 Halaman Index Sistem Pemesanan Jasa Travel Berbasis WAP
 Sumber : Analisis dan Perancangan

b. Tampilan Registrasi Anggota

Pada halaman ini, *user* pengunjung dapat mendaftarkan diri menjadi anggota agar dapat melakukan pemesanan jasa travel. *Field-field* yang harus diisi adalah: "nick user", "password", "nama user", "alamat", "no_ktp", "no.telp". Apabila semua data telah dimasukkan dengan lengkap pada *form* pendaftaran ini, maka data akan tersimpan pada *data store dt_anggota*. Dan apabila data yang diisi tidak lengkap, *nick user* atau nomor KTP yang dimasukkan telah digunakan, maka proses pendaftaran akan diulangi kembali.

BANNER	
NICK USER	: [.....]
PASSWORD	: [.....]
NAMA	: [.....]
ALAMAT	: [.....]
NO. KTP	: [.....]
NO. TELEPON	: [.....]
[kembali]	

FOOTER	
--------	--

Gambar 4.21 Halaman Daftar Anggota Baru Pengunjung
 Sumber : Analisis dan Perancangan

c. Tampilan Login Anggota

Menu ini merupakan halaman yang ditampilkan bagi *user* yang menekan *link Pesan Travel* pada halaman *index*. Apabila akan melakukan pemesanan travel, maka



field "Nick User" dan "Password" harus diisi terlebih dahulu untuk dapat menuju halaman berikutnya. Proses *login* akan gagal, dan akan memunculkan halaman kegagalan proses *login* apabila field "Nick User" dan "Password" diisi dengan data yang salah, atau data yang belum terdaftar pada sistem.

HEADER
JASA TIRTA PEMESANAN TRAVEL ONLINE
NICK USER : [....] PASSWORD : [.....]
[kembali]
FOOTER

Gambar 4.22 Halaman Login Anggota
 Sumber : Analisis dan Perancangan

d. Tampilan Konfirmasi Login Anggota

Menu ini merupakan halaman yang ditampilkan apabila *user* melakukan *login* dengan benar. Disediakan empat *link*: "Pesan Travel" untuk melakukan proses pemesanan travel, *link* "Batal Travel" untuk melakukan proses pembatalan pemesanan travel, *link* "Pesan Kurir" untuk melakukan proses pemesanan jasa antar barang dan *link* "Batal Kurir" untuk melakukan proses pembatalan pemesanan jasa antar barang.

HEADER
MENU ANGGOTA TRAVEL ONLINE
PEMESANAN TRAVEL ONLINE +Jadwal Keberangkatan +Pembatalan Pesanan +Lihat Pesanan +Logout
[kembali]
FOOTER

Gambar 4.23 Halaman Konfirmasi Login
 Sumber : Analisis dan Perancangan



e. **Tampilan Informasi Jasa Travel**

Halaman informasi transaksi ini akan muncul bila *link* "Lihat Jasa Travel " pada halaman *Index* ditekan. *User* yang menekan link kota tujuan, akan diarahkan menuju halaman informasi jasa travel. *User* anggota yang ingin memesan jasa travel harus *login* terlebih dahulu.

HEADER
JADWAL TRAVEL
KOTA TUJUAN :
Surabaya :
Tanggal : 7 Agustus 2009
Jam : 07.00 ; 10.00 ; 13.00
Tarif : Rp. 47.500
KOTA TUJUAN :
Kediri :
Tanggal : 7 Agustus 2009
Jam : 07.00 ; 10.00 ; 13.00
Tarif : Rp. 47.500
[kembali]
FOOTER

Gambar 4.24 Halaman Kedua Informasi Jasa Travel
Sumber : *Analisis dan Perancangan*

f. **Tampilan Profil Perusahaan**

Halaman profil perusahaan ini akan muncul bila *link* "profil perusahaan " pada halaman *Index* ditekan.

HEADER
SELAMAT DATANG DI PEMESANAN TRAVEL ONLINE
Penanggung Jawab :
Alamat Kantor :
No. Telp :
[kembali]
FOOTER

Gambar 4.25 Halaman Profil Perusahaan
Sumber : *Analisis dan Perancangan*

g. **Tampilan Informasi Transaksi**

Halaman informasi transaksi ini akan muncul bila *link* "Lihat Info Transaksi" pada halaman *Index* ditekan. *User* anggota yang ingin melihat informasi transaksi miliknya harus *login* terlebih dahulu.

HEADER	
JASA TIRTA PEMESANAN TRAVEL ONLINE	
NICK USER	: [....]
PASSWORD	: [.....]
[kembali]	
FOOTER	

Gambar 4.26 Halaman Pertama Informasi Transaksi
Sumber : *Analisis dan Perancangan*

HEADER	
Kode Transaksi	:
Kota Tujuan	:
Jenis Pesan	:
Tanggal Berangkat	:
Jam Berangkat	:
Alamat Penjemputan	:
Alamat Tujuan	:
[logout]	
FOOTER	

Gambar 4.27 Halaman Kedua Informasi Transaksi
Sumber : *Analisis dan Perancangan*

4.2.5.2 Antarmuka User Admin

User admin merupakan *user* yang dapat bertindak sebagai pengelola data yang ada dari masing-masing *data store* yang berhubungan dengan data-data tiket.

a. **Tampilan Login Admin, Customer Service, dan Manajemen**

Menu ini merupakan halaman *login* bagi *user* admin, customer service, dan manajemen. Admin, customer service, dan manajemen mempunyai nama *user* dan *password* yang berbeda, yang akan menuju ke halaman yang berbeda pula.

BANNER	
Login Administrator & Manajemen	
Login	<input type="text"/>
Password	<input type="text"/>
<input type="button" value="Login"/>	
FOOTER	

Gambar 4.28 Halaman Login Admin, Customer Service, dan Manajemen
 Sumber : Analisis dan Perancangan

b. Tampilan Menu Utama Customer Service

Menu ini merupakan menu utama untuk *user* customer service. Berisi *link-link* yang dapat dipilih untuk memasukkan data baru atau mengedit data yang sudah ada sebelumnya.

HEADER		<input type="button" value="logout"/>
Edit Data :		
Data Anggota		
Data Mobil		
Jadwal Keberangkatan		
Data Tiket		
FOOTER		

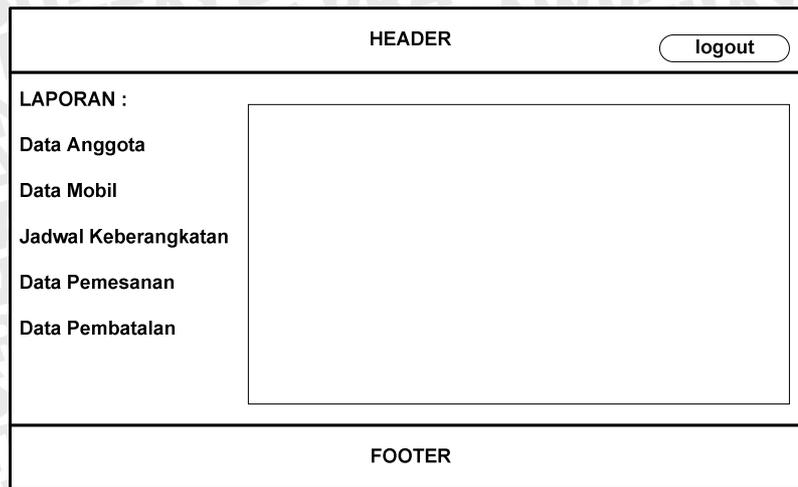
Gambar 4.29 Halaman Masukkan dan Edit Data
 Sumber : Analisis dan Perancangan

4.2.5.3 Antarmuka User Manajemen

User manajemen merupakan *user* dengan hak akses sebagai pihak manajer. Manajemen dapat melihat laporan-laporan dari sistem pemesanan Jasa Travel ini.

a. Tampilan Menu Utama Manajemen

Menu ini merupakan menu utama bagi *user* manajemen setelah melakukan *login* sebagai manajemen. Berisi *link-link* yang dapat dipilih untuk melihat laporan-laporan yang ada.



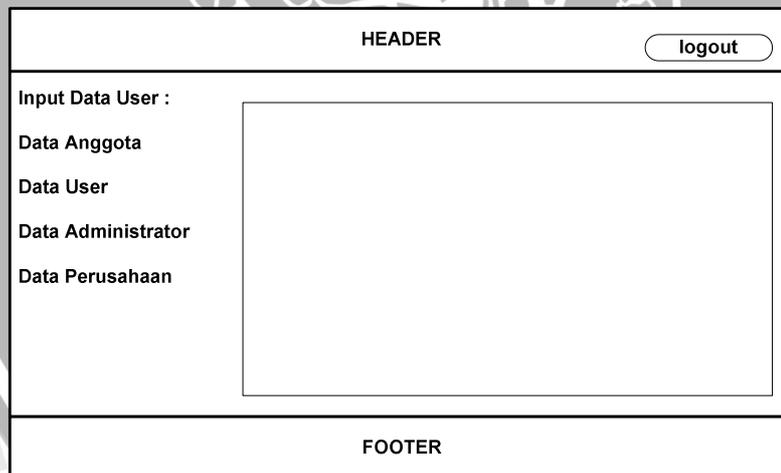
Gambar 4.30 Halaman Lihat Laporan
 Sumber : Analisis dan Perancangan

4.2.5.4 Antarmuka User Admin Perangkat Lunak

User admin merupakan admin yang bertindak sebagai pengelola data pada *data store* yang berhubungan dengan hak akses *user* atau registrasi anggota.

a. Tampilan Menu Utama Admin

Menu ini merupakan menu utama bagi *user* admin setelah melakukan *login* sebagai admin. Berisi *link-link* yang dapat dipilih untuk mengelola data *user*.



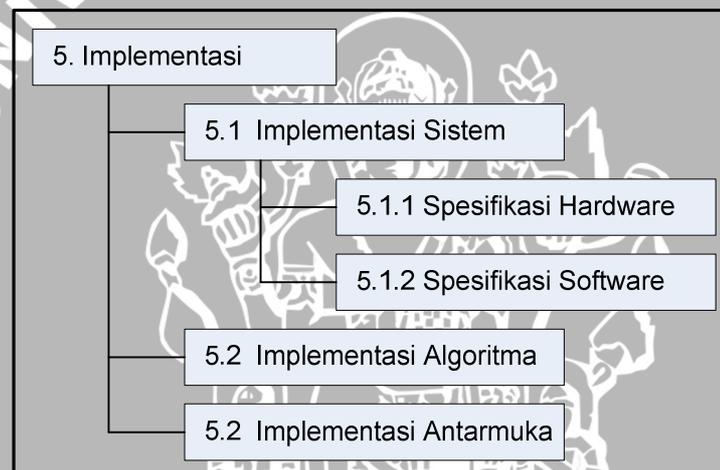
Gambar 4.31 Halaman Menu Utama Admin
 Sumber : Analisis dan Perancangan

BAB V

IMPLEMENTASI

Bab ini membahas tentang implementasi perangkat lunak Sistem Pemesanan Jasa Travel menggunakan layanan WAP berdasarkan hasil yang telah didapatkan dari analisis kebutuhan dan proses perancangan perangkat lunak. Bab ini terdiri dari pembahasan implementasi algoritma dan implementasi antarmuka, ditambah dengan penjelasan tentang spesifikasi lingkungan dimana sistem diimplementasikan.

Implementasi secara keseluruhan dapat digambarkan dengan diagram pohon seperti yang ditunjukkan dalam Gambar 5.1 berikut:



Gambar 5.1 Diagram Pohon Implementasi Sistem
Sumber : [Implementasi]

5.1 Implementasi Sistem

Dari hasil analisis kebutuhan dan perancangan yang telah diuraikan pada Bab 4, dilakukan implementasi menjadi sebuah sistem Pemesanan Jasa Travel yang nyata agar bisa berfungsi sesuai dengan kebutuhan. Sistem diimplementasikan pada spesifikasi *hardware* dan *software*.

5.1.1 Spesifikasi Hardware

Sistem Pemesanan Jasa Travel dengan layanan WAP diimplementasikan dengan menggunakan bahasa pemrograman web PHP, *database* MySQL dan *Apache* sebagai *web server*. Sistem diimplementasikan dengan menggunakan spesifikasi sebagai berikut:

1. Konfigurasi server

Spesifikasi *hardware* minimal:

- CPU Intel Pentium III 800 MHz
- Memori 256 Mb SD RAM

2. Konfigurasi *browser*

- Tipe ponsel : LG Shine
- Menggunakan jaringan GPRS dengan operator Indosat IM3

5.1.2 Spesifikasi Software

Pengembangan Sistem Pemesanan Jasa Travel dengan layanan WAP menggunakan perangkat lunak dengan spesifikasi sebagai berikut :

Spesifikasi *software* :

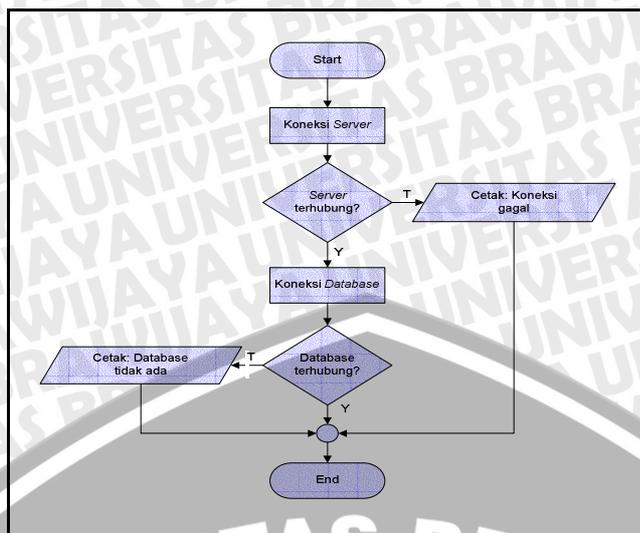
- Microsoft Windows XP Profesional Version 2002 Service Pack 2
- PHP Versi 5.1.4
- MySQL Versi 5.0.22
- Apache *Web Server* Versi 2.2.2
- Emulator WAP *m3gate* Versi 0.5

5.2 Implementasi Algoritma

Penyajian algoritma yang digunakan berupa tulisan dengan menggunakan metode *pseudocode*. Metode *pseudocode* merupakan metode penyajian algoritma dengan membuat kode yang mirip dengan kode pemrograman yang sesungguhnya.

5.2.1 Proses Koneksi

Proses koneksi dimulai dengan permintaan koneksi pada server. Apabila berhasil koneksi maka akan diteruskan dengan permintaan untuk memilih *database* dalam hal ini adalah *database* *traveltime*. Proses koneksi tersebut ditunjukkan pada *flowchart* dalam Gambar 5.2.



Gambar 5.2 Algoritma Koneksi
Sumber: Implementasi

Pseudocode dari algoritma proses Koneksi adalah sebagai berikut :

```

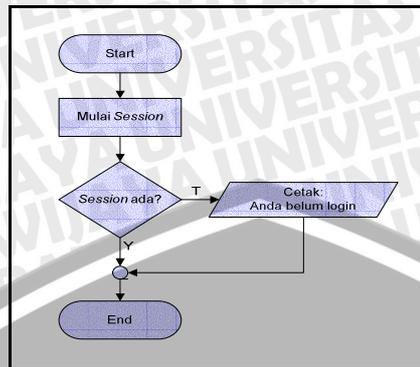
1 | PROCEDURE koneksi
2 |   Prosedur pengecekan koneksi dengan server. Dan
   prosedur pemilihan database yg akan digunakan jika
   proses koneksi berhasil
3 |   CONNECT TO server
4 |   INPUT host and user and password
5 |   CONNECT TO database
6 |   IF database connected
7 |     SELECT database
8 |   ELSE
9 |     PRINT "koneksi database gagal"
10 |
11 |   ENDIF
12 | ENDPROC
  
```

Gambar 5.3 Algoritma Proses Koneksi
Sumber : [Implementasi]

5.2.2 Proses Cek Session

Proses Cek *Session* digunakan untuk membatasi *user* dalam mengakses halaman yang tersedia yaitu dimulai dengan proses mulai *session*. Apabila *user* tidak memiliki hak sebagai admin, maka *user* tidak bisa mengakses halaman-halaman admin begitu juga sebaliknya apabila *user* tidak memiliki hak sebagai anggota, maka *user* tidak bisa mengakses halaman-halaman anggota. *User* yang tidak memiliki hak apapun untuk melakukan *login* akan berstatus sebagai pengunjung. Proses cek *session* pada sistem ini

terbagi empat, yaitu proses cek *session user* admin, customer service, manajemen, dan anggota. Proses - proses tersebut ditunjukkan pada *flowchart* dalam Gambar 5.3.



Gambar 5.4 Algoritma Cek *Session* Untuk Anggota
Sumber: Implementasi

Pseudocode dari algoritma proses *Session* adalah sebagai berikut :

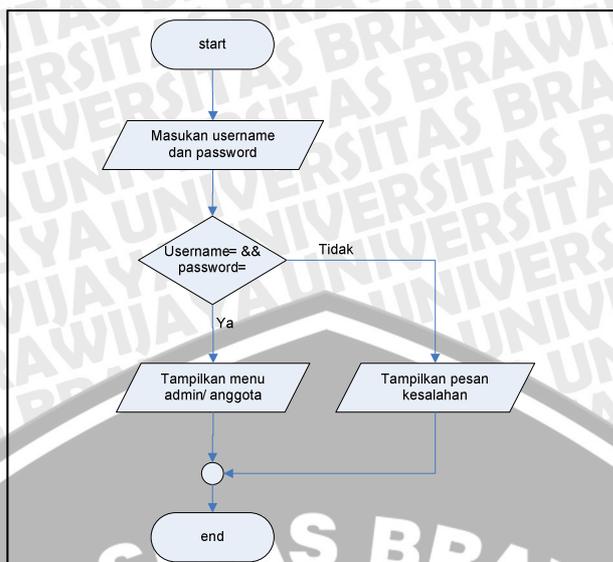
```

1 |   PROCEDURE session
2 |     Prosedur pengecekan untuk membatasi user dalam
      mengakses halaman yang tersedia
3 |   START session
4 |   TYPE user, password IS STRING
5 |   INPUT user_value and password_value
6 |   CONNECT TO database
7 |   IF (username=user_value and
8 |     password=password_value) THEN
9 |     SESSION REGISTER as user_value and
      password_value
10 |  ELSE
11 |    PRINT "Anda belum login, silahkan login
      terlebih dahulu"
13 |  ENDIF
14 |  ENDPROC
  
```

Gambar 5.5 Algoritma Proses *Session*
Sumber : [Implementasi]

5.2.3 Proses Login

Pada proses *login* terdapat empat login, yaitu *login* admin, *login* customer service, *login* manajemen dan *login* user. Untuk proses *login* admin, dimulai dengan memberikan masukan nilai variabel *username* dan *password* kepada sistem dan dilanjutkan dengan seleksi tabel *dt_admin*. Apabila data sesuai, maka sistem akan melakukan registrasi *session*. Jika semua kondisi terpenuhi maka sistem akan menampilkan menu admin. Proses *login* admin tersebut sama dengan apa yang terjadi pada proses *login user* yang lainnya. Proses *login* admin ditunjukkan pada *flowchart* dalam Gambar 5.4.



Gambar 5.6 Algoritma Validasi Login

Sumber: Implementasi

Pseudocode dari algoritma proses login adalah sebagai berikut :

```

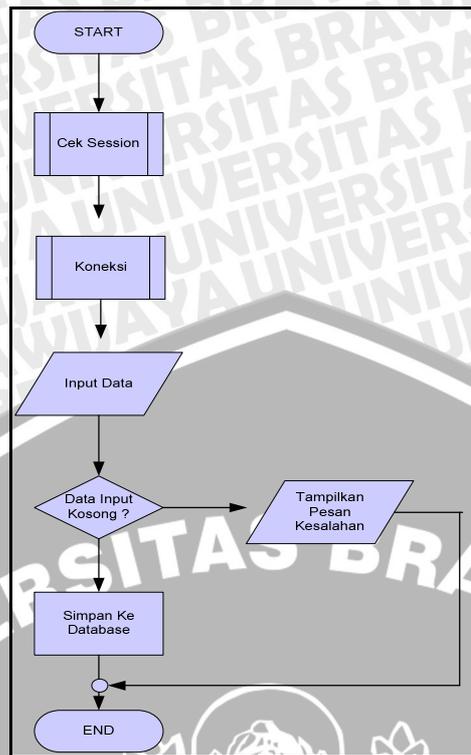
1 | PROCEDURE login
2 |     Prosedur pengecekan login serta untuk menampilkan
   |     menu yang dapat diakses anggota
3 |     TYPE user, passwd IS STRING
4 |     INPUT user_value and passwd_value
5 |     CONNECT TO database
6 |     SELECT username,password FROM database WHERE
   |     username='$user' AND password='$passwd'
7 |     IF (username=user_value and
   |     password=password_value) THEN
8 |         SESSION REGISTER as user_value and
   |         password_value
9 |         PRINT "Selamat Datang"
10 |        VIEW menu utama
11 |     ELSE
12 |        PRINT "Login Anda Salah"
13 |     ENDIF
14 |     ENDPROC
    
```

Gambar 5.7 Algoritma Proses Login

Sumber : [Implementasi]

5.2.4 Proses Input Data

Proses ini adalah proses yang dilakukan oleh admin yang merupakan proses untuk memasukkan data-data jasa travel ke dalam *database*. Untuk proses *input* data, dimulai dengan proses cek *session* dan proses koneksi. Kemudian dilanjutkan dengan mengisi data-data. Jika semua kondisi terpenuhi maka sistem akan menyimpan data-data tersebut ke dalam tabel yang telah ditentukan. Proses *input* data ditunjukkan pada *flowchart* dalam Gambar 5.5.



Gambar 5.8 Algoritma *Input Data*
 Sumber: *Implementasi*

Pseudocode dari algoritma proses input data adalah sebagai berikut :

```

1 |   PROCEDURE input data
2 |       prosedur untuk memasukkan data kedalam
3 |       database
4 |   START session
5 |   CONNECT TO server
6 |   TYPE id, nama_data, isi_data IS STRING
7 |   INPUT nama_data value and isi_data_value
8 |   CONNECT TO database
9 |   SOL INSERT INTO database VALUES (id_value,
10 |      nama_data_value and isi_data_value)
11 |   IF (sukses) THEN
12 |       simpan
13 |   ELSE
14 |       PRINT "Data Input Gagal"
15 |   ENDIF
16 |   ENDPROC
  
```

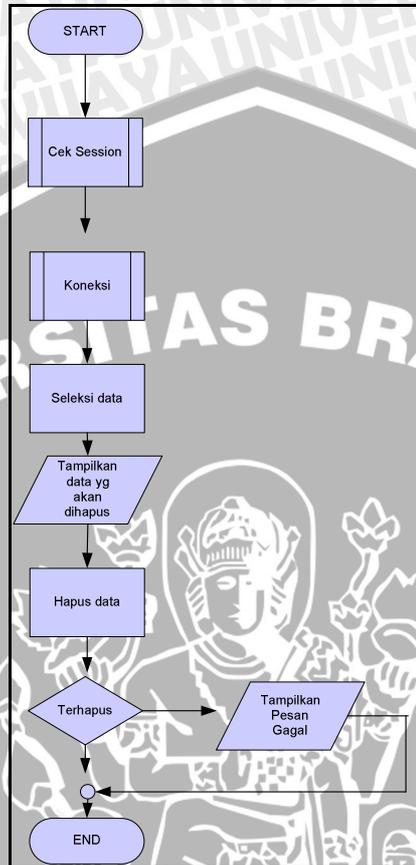
Gambar 5.9 Algoritma *Proses Input Data*
 Sumber : [*Implementasi*]

5.2.5 Proses Hapus Data

Proses ini adalah proses yang dilakukan oleh admin yang merupakan proses untuk menghapus data-data dari *database*. Untuk proses hapus data , dimulai dengan proses cek *session* dan proses koneksi. Kemudian dilanjutkan dengan seleksi tabel data.



Setelah data , selanjutnya dilakukan penghapusan data sesuai dengan data yang diseleksi. Jika semua kondisi terpenuhi maka sistem akan menghapus data tersebut dari tabel yang bersangkutan. Proses hapus data bus ditunjukkan pada *flowchart* dalam Gambar 5.9.



Gambar 5.10 Algoritma Hapus Data
Sumber: *Implementasi*

Pseudocode dari algoritma proses hapus data adalah sebagai berikut :

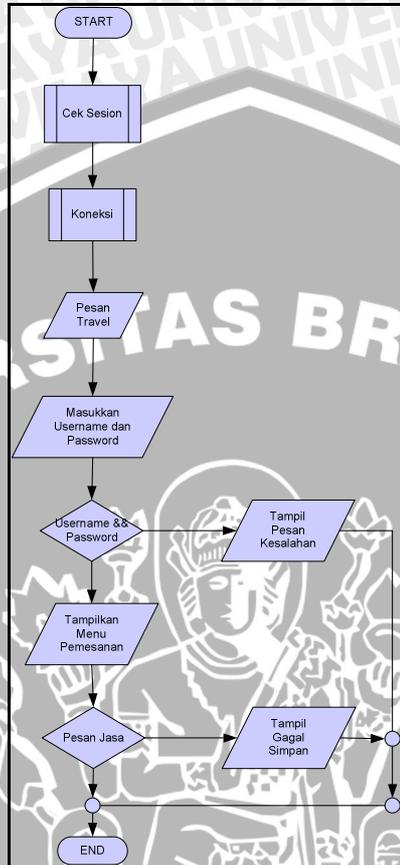
```

1 |   PROCEDURE hapus data
2 |       prosedur untuk menghapus data dari database
3 |   START session
4 |   CONNECT TO server
5 |   TYPE id   IS STRING
6 |   CONNECT TO database
7 |   IF (GET (id) )
8 |       SQL DELETE FROM database WHERE id = (GET (id) )
9 |   IF (delete succes) THEN
10 |       PRINT "data telah dihapus";
11 |       ELSE
12 |       PRINT "Data gagal dihapus";
13 |       ENDIF
14 |   ENDIF
15 |   ENDPROC
  
```

Gambar 5.11 Algoritma Hapus Data
Sumber : [*Implementasi*]

5.2.6 Proses Transaksi Pemesanan

Proses transaksi merupakan proses yang dilakukan oleh anggota dalam melakukan proses pemesanan travel. Terbagi menjadi proses transaksi pemesanan dan proses transaksi pembatalan.



Gambar 5.12 Algoritma Transaksi Pemesanan

Sumber: *Implementasi*

Pseudocode dari algoritma proses transaksi pemesanan adalah sebagai berikut :

```
1 | PROCEDURE transaksi
2 |     Prosedur pemesanan jasa melalui menu yang
   |     dapat diakses oleh anggota
3 |
4 |     load module "inc/sesi.php";
5 |     load module"inc/header_wml.php";
6 |     load module"inc/koneksi.php";
7 |
8 |     SET sql to "select jumlah from tr_counter where
   |     jenis='$jns'"
9 |     SET queri TO mysql_query($sql,$konektor)
10 |    SET row TO mysql_fetch_row($queri)
11 |    SET add_id TO $row[0] + 1
12 |    SET sql = "select sisa_tiket FROM tr_tiket where
   |    id_tiket='$stk'"
13 |    SET queri = mysql_query($sql,$konektor)
14 |    SET row = mysql_fetch_row($queri)
```

```

13| SET sisa_tiket = $row[0]
14| SET tgl_pesan = date("Y-m-d")
15| SET jam_pesan = date("H:i:s")
16| IF jns == "travel" THEN
17|     {
18|         SET id_pesan = "TR0".$add_id
19|     }
20| IF ($jns == "kurir") THEN {
21|     SET id_pesan = "KR0".$add_id
22| }
23| SET queri = mysql_query($sql,$konektor)
24| IF queri THEN
25|     SET sql2 = update tr_counter SET jumlah='$add_id'
26|     WHERE jenis='$jns'
27|     SET queri2 = mysql_query($sql2,$konektor)
28|     SET add_sisa_tiket = $sisa_tiket - $jml
29|     SET sql2 = update tr_tiket set
30|     sisa_tiket='$add_sisa_tiket' where id_tiket='$tk'
31|     SET queri2 = mysql_query($sql2,$konektor)
32|     PRINT "Pesan telah berhasil tersimpan !<br/>Nomor
33|     Pemesanan anda adalah :
34|     <b>$id_pesan</b><br/>"
35|     PRINT "[&nbsp;<a href=\"menu_anggota.php\">Menu
36|     Anggota</a>&nbsp;]"
37| }
38| ELSE THEN
39|     PRINT "Pesan Gagal Tersimpan tersimpan
40|     !<br/>Silahkan hubungi customer service
41|     kami<br/>"
42|     PRINT "[&nbsp;<a href=\"menu_anggota.php\">Menu
43|     Anggota</a>&nbsp;]"
44| }
45| END IF

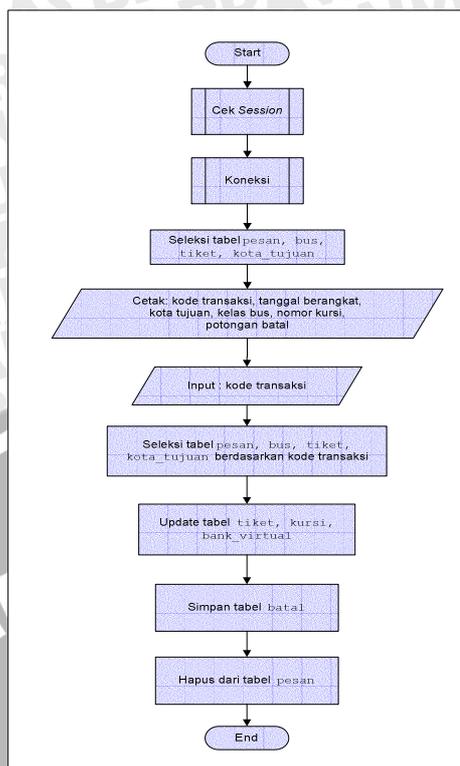
```

Gambar 5.13 Algoritma Transaksi Pemesanan
Sumber : [Implementasi]

5.2.7 Proses Transaksi Pembatalan Tiket

Proses ini adalah proses yang dilakukan oleh anggota yang merupakan proses transaksi pembatalan tiket yang telah dipesan oleh anggota. Berdasarkan hasil seleksi akan ditampilkan sejumlah kode transaksi yang masih berlaku yang dapat dipilih untuk dibatalkan oleh pemesan.

Apabila salah satu kode transaksi dipilih, akan dilakukan seleksi tabel berdasarkan kode transaksi yang telah dipilih tersebut. Dan juga akan melakukan penyimpanan ke tabel batal.



Gambar 5.14 Algoritma Transaksi Pembatalan Tiket
 Sumber: *Implementasi*

Pseudocode dari algoritma proses pembatalan transaksi adalah sebagai berikut :

```

1 |   PROCEDURE pembatalan transaksi
2 |       Prosedur pembatalan jasa melalui menu yang
          dapat diakses oleh anggota berdasarkan kode
          pemesanan
3 |   LOAD MODULE inc/sesi.php
4 |   LOAD MODULE inc/header_wml.php
5 |   LOAD MODULE inc/koneksi.php
6 |
7 |   SET nomor to nomor
8 |   SET tgl to tgl
9 |   PRINT "<wml">
10 |  PRINT "load module"
11 |  SET sql2 to "update tr_pemesanan set status='0' WHERE
          id_pemesanan='\$nomor'"
12 |  SET query2 to "mysql_query(\$sql2, \$konektor)"
13 |  IF query2 THEN
14 |      PRINT "Pesanan dengan Nomor Pemesanan :<b> \$nomor</b>
          telah dibatalkan <br/>"
15 |      PRINT "[&nbsp;<a href='\"menu_anggota.php\">Menu
          Anggota</a>&nbsp;<\/p>"
          ELSE THEN
17 |      PRINT "Pesanan dengan Nomor Pemesanan :<b> \$nomor</b>
          tidak dapat dibatalkan <br/>Silahkan menghubungi
          Customer Service kami<br/>";
18 |      PRINT "[&nbsp;<a href='\"menu_anggota.php\">Menu
          Anggota</a>&nbsp;<\/p>"
19 |  END IF
20 |  PRINT "<\/p>"
21 |  PRINT "<\/card>"
22 |  PRINT "<\/wml>"
    
```



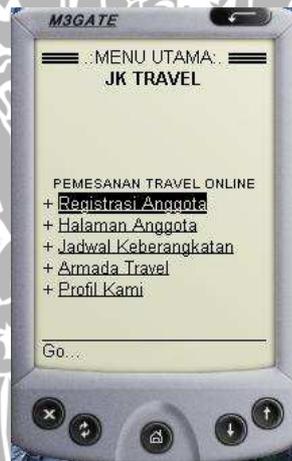
Gambar 5.15 Algoritma Pembatalan Transaksi
Sumber : [Implementasi]

5.3 Implementasi Antarmuka

Implementasi antarmuka website dan WAP site merupakan hasil dari perancangan antarmuka yang terdapat pada sub bab 4..

5.3.1 Implementasi Antarmuka WAP Anggota dan Pengunjung

Menu ini merupakan halaman yang pertama kali diakses *user* pada saat alamat URL situs ini diakses melalui telepon genggam. Pada halaman ini terdapat lima *link*, yaitu: Registrasi Anggota, Halaman Anggota, Jadwal Keberangkatan, Armada Travel dan Profil Jasa Travel. *Link* yang dapat diakses oleh *user* pengunjung adalah Registrasi Anggota, Jadwal Keberangkatan dan Profil Jasa Travel. Implementasi halaman WAP anggota dan pengunjung ditunjukkan pada gambar 5.8



Gambar 5.16 Implementasi Halaman Utama
Sumber : [Implementasi]

5.3.2 Implementasi Login

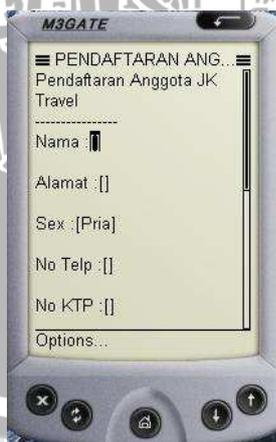
Anggota yang ingin memesan jasa travel harus login terlebih dahulu sebelum menggunakan aplikasi yang terdapat pada wap-site. Implementasi form login diunjukkan dalam Gambar 5.9.



Gambar 5.17 Implementasi Halaman Login Anggota
Sumber : [Implementasi]

5.3.3 Implementasi Registrasi Anggota

Pada halaman ini, *user* pengunjung dapat mendaftarkan diri menjadi anggota agar dapat melakukan pemesanan jasa travel. *Field-field* yang harus diisi adalah: "nick user", "password", "nama user", "alamat", "no_ktp", "no.telp". Apabila semua data telah dimasukkan dengan lengkap pada *form* pendaftaran ini, maka data akan tersimpan pada *data store dt_anggota*. Dan apabila data yang diisi tidak lengkap, *nick user* atau nomor KTP yang dimasukkan telah digunakan, maka proses pendaftaran akan diulangi kembali. Implementasi halaman registrasi ditunjukkan dalam gambar 5.10



Gambar 5.18 Implementasi Halaman Registrasi Anggota
Sumber : [Implementasi]

5.3.4 Implementasi Jadwal Keberangkatan

Halaman informasi Jadwal Keberangkatan ini akan muncul bila *link* "Jadwal Keberangkatan " pada halaman *Index* ditekan. *User* yang menekan link pesan

travel, akan diarahkan menuju halaman pemesanan jasa travel. *User* anggota yang ingin memesan jasa travel harus *login* terlebih dahulu. Implementasi halaman jadwal keberangkatan ditunjukkan dalam gambar 5.11



Gambar 5.19 Implementasi Halaman Jadwal Keberangkatan
Sumber : [Implementasi]

5.3.5 Implementasi Halaman Menu Anggota

Menu ini merupakan halaman yang ditampilkan apabila *user* melakukan *login* dengan benar. Disediakan empat *link*: "Jadwal Keberangkatan" untuk melakukan proses pemesanan travel, *link* "Pembatalan Pesanan" untuk melakukan proses pembatalan pemesanan travel, *link* "Lihat Pesanan" untuk melihat informasi transaksi dan *link* "Logout" untuk keluar dari menu anggota. Implementasi halaman menu anggota ditunjukkan dalam gambar 5.12



Gambar 5.20 Implementasi Halaman Menu Anggota
Sumber : [Implementasi]

5.3.6 Implementasi Halaman Konfirmasi Pemesanan

Halaman ini merupakan halaman yang akan ditampilkan setelah anggota melakukan pemesanan jasa. Anggota harus melakukan konfirmasi pemesanannya, jika tidak maka anggota dianggap tidak memesan jasa apapun. Implementasi halaman konfirmasi pemesanan ditunjukkan dalam gambar 5.13



Gambar 5.21 Implementasi Halaman Konfirmasi Pemesanan
Sumber : [Implementasi]

5.3.7 Implementasi Halaman Detail Pesanan

Halaman ini akan ditampilkan jika user yang telah melakukan konfirmasi pemesanan ingin melihat kembali detail dari pemesanan jasa yang telah dilakukan. Implementasi halaman detail pesanan ditunjukkan dalam gambar 5.14



Gambar 5.22 Implementasi Halaman Detail Pesanan
Sumber : [Implementasi]

5.3.8 Implementasi Antarmuka WEB *user* admin dan manajemen

Menu ini merupakan halaman yang pertama kali diakses *user* admin dan manajemen pada saat alamat URL situs ini diakses melalui komputer. Halaman ini

digunakan oleh *user* admin dan manajemen untuk mengedit, memasukkan, menghapus data-data dan melihat laporan-laporan yang berhubungan dengan sistem pemesanan jasa travel. Implementasi halaman WEB *user* admin dan manajemen ditunjukkan pada gambar 5.23

Gambar 5.23 Implementasi Fasilitas Login *user* admin dan manajemen

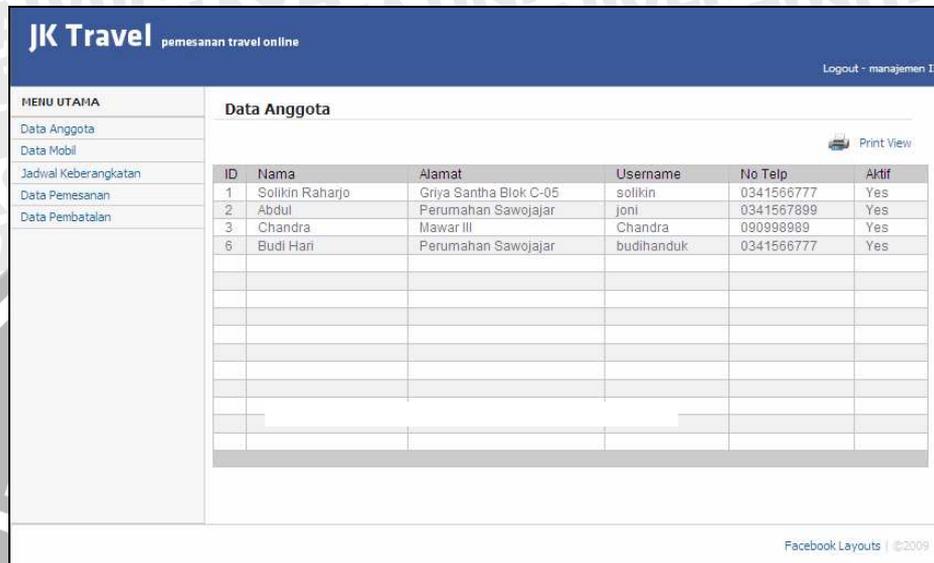
Apabila login *customer service* valid, maka akan masuk ke halaman menu utama cs yang mana dalam halaman tersebut terdapat pilihan untuk proses input data dan edit data. Implementasi menu utama *customer service* ditunjukkan dalam Gambar 5.24.

ID	Nama	Alamat	Username	No Telp	Aktif
1	Solikin Raharjo	Griya Santha Blok C-05	solikin	0341566777	Yes
2	Abdul	Perumahan Sawojajar	joni	0341567899	Yes
3	Chandra	Mawar III	Chandra	090998989	Yes
6	Budi Hari	Perumahan Sawojajar	budihanduk	0341566777	Yes

Gambar 5.24 Halaman Edit dan Input Data *Customer Service*

5.3.9 Implementasi Antarmuka Manajemen

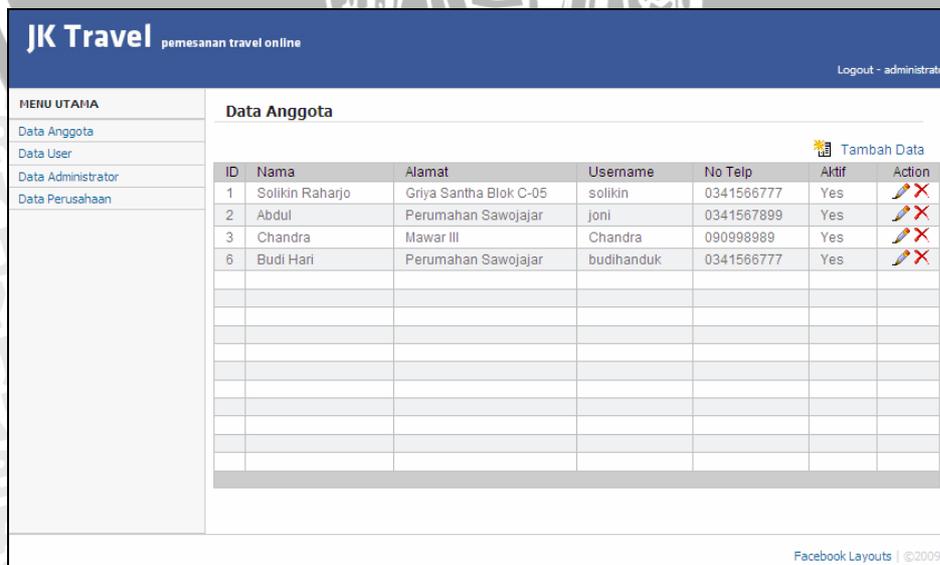
Apabila melakukan login sebagai manajemen, maka akan masuk ke antarmuka untuk manajemen. Pada menu utama ini, manajemen dapat memilih untuk melihat laporan yang telah disediakan. Implementasi menu utama manajemen ditunjukkan dalam Gambar 5.25



Gambar 5.25 Halaman Antarmuka Manajemen

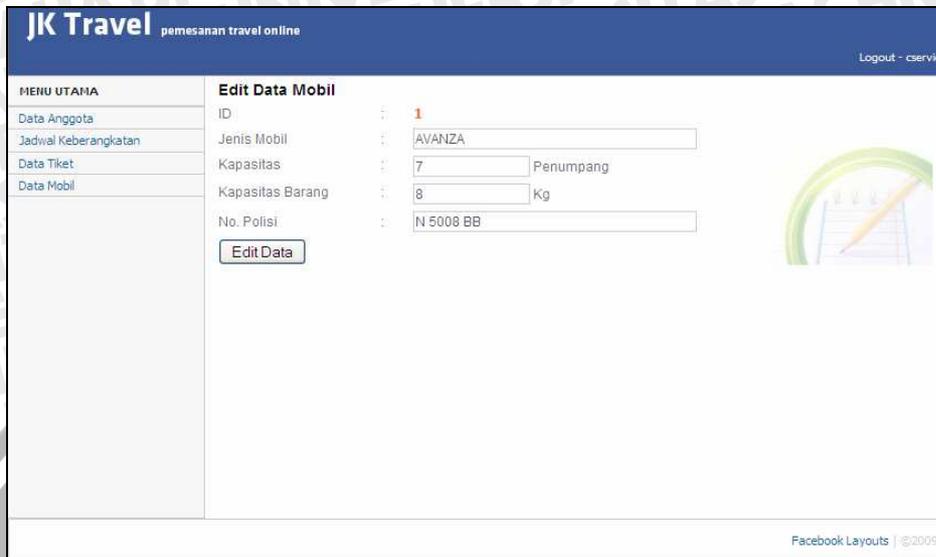
5.3.10 Implementasi Antarmuka Admin

Apabila melakukan login sebagai admin, maka akan masuk ke antarmuka untuk admin. Pada menu utama ini, admin dapat memilih proses input atau proses edit data yang telah disediakan. Implementasi menu utama admin ditunjukkan dalam Gambar 5.26



5.3.11 Implementasi Antarmuka Edit Data

Halaman ini ditampilkan apabila *user* login sebagai *customer service*, pada menu ini, *customer service* dapat melakukan editing terhadap data yang sudah ada.

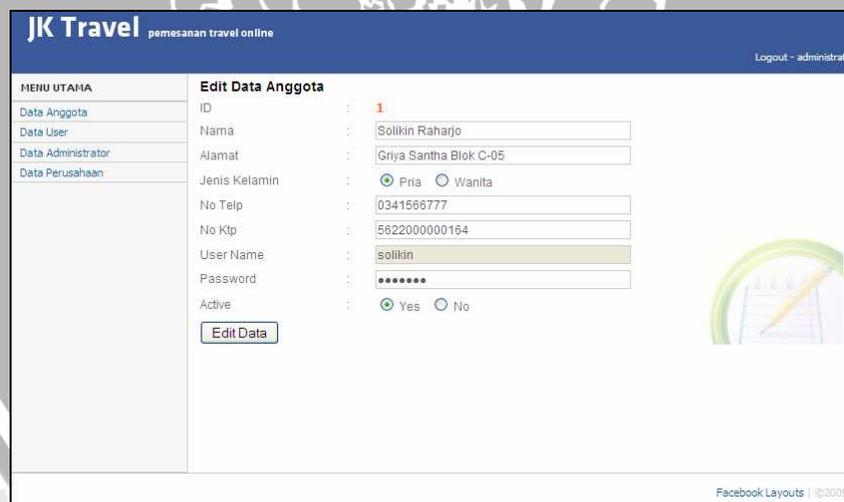


MENU UTAMA	Edit Data Mobil
Data Anggota	ID : 1
Jadwal Keberangkatan	Jenis Mobil : AVANZA
Data Tiket	Kapasitas : 7 Penumpang
Data Mobil	Kapasitas Barang : 8 Kg
	No. Polisi : N 5008 BB
	<input type="button" value="Edit Data"/>

Gambar 5.27 Halaman Edit Data Anggota

5.3.12 Implementasi Antarmuka Edit Data Anggota

Halaman ini ditampilkan saat *user* login sebagai admin dan ingin mengedit data anggota yang sudah terdaftar di sistem pemesanan jasa travel. Implementasi menu edit data anggota ditunjukkan dalam gambar 5.28



MENU UTAMA	Edit Data Anggota
Data Anggota	ID : 1
Data User	Nama : Solikin Raharjo
Data Administrator	Alamat : Griya Santha Blok C-05
Data Perusahaan	Jenis Kelamin : <input checked="" type="radio"/> Pria <input type="radio"/> Wanita
	No Telp : 0341566777
	No Ktp : 562200000164
	User Name : solikin
	Password : *****
	Active : <input checked="" type="radio"/> Yes <input type="radio"/> No
	<input type="button" value="Edit Data"/>

Gambar 5.28 Halaman Edit Data Anggota

5.3.13 Implementasi Antarmuka Lihat Laporan

Halaman ini akan ditampilkan jika *user* login sebagai manajemen, sehingga bisa mengakses laporan yang ada dari sistem pemesanan jasa travel. Implementasi menu lihat laporan ditunjukkan dalam gambar 5.29

JK Travel pemesanan travel online Logout - manajemen II

Menu Utama
 Data Anggota
 Data Mobil
 Jadwal Keberangkatan
 Data Pemesanan
 Data Pembatalan

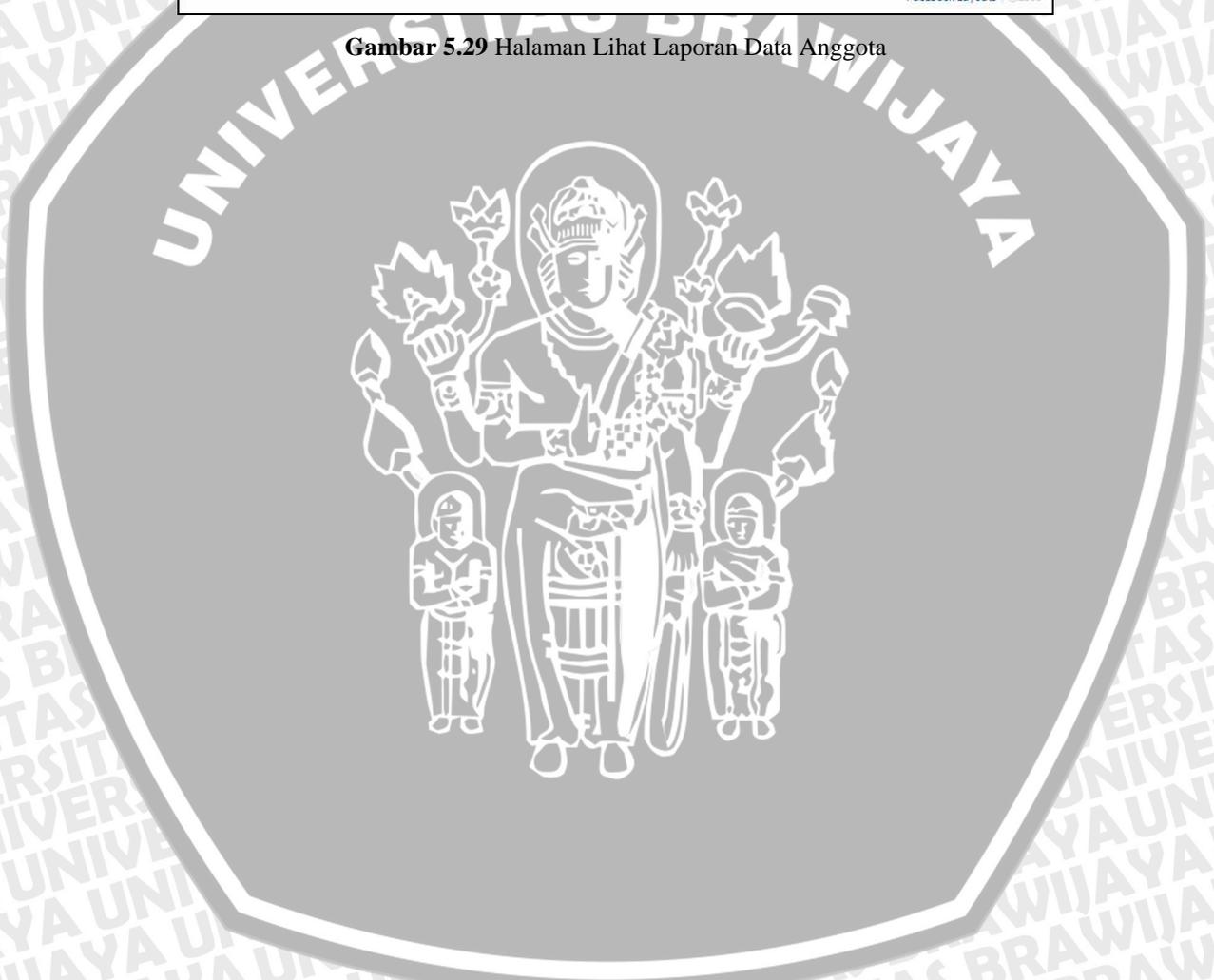
Data Pemesanan Print View

Jenis Pesan : Semua

ID	Kota Tujuan	Tgl Berangkat	Jam Berangkat	tgl pesan	Jenis Mobil	Customer	Jumlah Pesan	Harga Total
KR02	Madiun	2009-07-20	08.00	2009-07-30	kurir	solikin	0	Rp. 40000
KR03	Surabaya	2009-07-19	15.00	2009-08-04	kurir	solikin	0	Rp. 20000
KR04	Madiun	2009-07-20	08.00	2009-08-05	kurir	Chandra	0	Rp. 40000
KR05	Surabaya	2009-07-20	07.00	2009-08-05	kurir	joni	2	Rp. 40000
KR06	Surabaya	2009-07-20	07.00	2009-08-05	kurir	joni	2	Rp. 40000
KR07	Surabaya	2009-07-20	07.00	2009-08-05	kurir	joni	0	Rp. 20000
KR08	Surabaya	2009-07-19	18.00	2009-08-05	kurir	joni	2	Rp. 40000
TR01	Surabaya	2009-07-20	07.00	2009-07-27	travel	solikin	1	Rp. 120000
TR013	Surabaya	2009-07-20	12.00	2009-08-05	travel	joni	1	Rp. 40000
TR02	Surabaya	2009-07-19	07.00	2009-07-27	travel	solikin	1	Rp. 80000
TR03	Surabaya	2009-07-19	07.00	2009-07-27	travel	solikin	2	Rp. 80000
TR04	Madiun	2009-07-20	17.00	2009-07-27	travel	solikin	4	Rp. 320000
TR05	Surabaya	2009-07-19	07.00	2009-07-28	travel	joni	2	Rp. 80000
TR07	Surabaya	2009-07-19	07.00	2009-07-28	travel	joni	2	Rp. 80000
TR09	Surabaya	2009-07-20	12.00	2009-07-30	travel	solikin	2	Rp. 80000

Facebook Layouts | ©2009

Gambar 5.29 Halaman Lihat Laporan Data Anggota



BAB VI PENGUJIAN

Bab ini membahas proses pengujian terhadap perangkat lunak Sistem Pemesanan Jasa Travel dengan layanan WAP yang telah dibangun. Proses pengujian dilakukan melalui dua tahapan (strategi) yaitu pengujian algoritma dan pengujian validasi. Pengujian algoritma menggunakan teknik pengujian *white box* (*white box Testing*). Pengujian validasi menggunakan teknik pengujian *black box* (*black box Testing*).

6.1 Pengujian Algoritma

Teknik pengujian yang dilakukan adalah teknik pengujian kotak putih (*white box testing*) dengan teknik pengujian jalur dasar (*basis path testing*). Pada pengujian jalur dasar akan dilakukan pembuatan *programme design language* (PDL), pembuatan grafik alir (*flow graph*), menentukan kompleksitas siklomatis, menentukan jalur independen dan melakukan kasus uji.

6.1.1 Pengujian Proses Login

1. Pembuatan PDL untuk desain *test case* dengan simpul – simpul yang diidentifikasi

```

PROCEDURE login

Prosedur pengecekan login serta untuk menampilkan menu yang dapat diakses user
TYPE user, password IS STRING

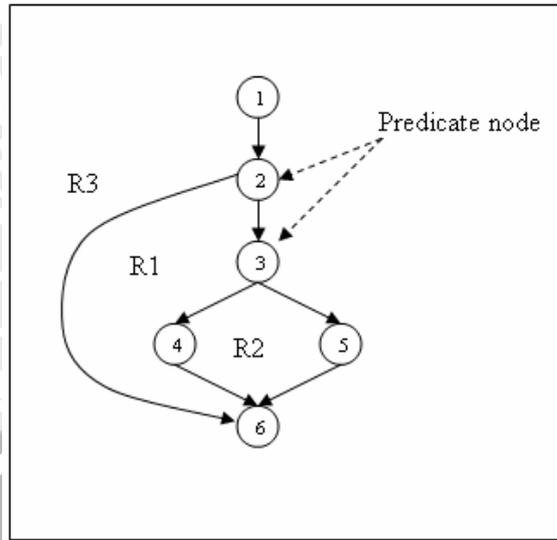
1 { INPUT user_value and password_value
  CONNECT TO database
  SELECT username,password FROM database WHERE username='$user' AND
  password='$password'
2 ← IF (username=user_value and
3 ← password=password_value) THEN
4 { SESSION REGISTER as user_value and password_value
  PRINT "login berhasil"
  VIEW menu
5 { ELSE
  PRINT "Login Tidak Berhasil"
6 ← ENDF
ENDPROC
    
```

Gambar 6.1 PDL Proses Login

Sumber : [Pengujian]

2. Pembuatan grafik alir (*flow graph*) yang sesuai





Gambar 6.2 Flow Graph Proses login

Sumber : [Pengujian]

3. Penentuan kompleksitas siklomatis dari grafik alir resultan

$$V(G) = 3 \text{ region}$$

$$V(G) = 7 \text{ edge} - 6 \text{ node} + 2 = 3$$

$$V(G) = 2 \text{ predicate node} + 1 = 3$$

4. Penentuan jalur independen

Jalur 1 : 1 – 2 – 6

Jalur 2 : 1 – 2 – 3 – 5 – 6

Jalur 3 : 1 – 2 – 3 – 4 – 6

5. Melakukan kasus uji

Penentuan kasus uji untuk masing-masing jalur dan hasil eksekusi untuk masing-masing kasus uji dijelaskan pada Tabel 6.1.

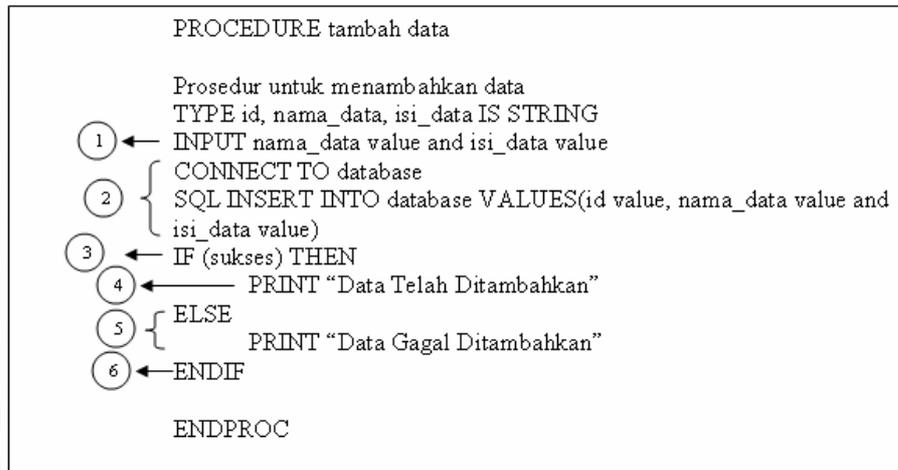
Tabel 6.1 Kasus uji untuk pengujian proses login

Jalur	Kasus uji	Hasil yang diharapkan	Hasil yang didapatkan
1	username = null or username != username value	User gagal login dan keluar pesan error	User gagal login
2	password = null or password != password value	User gagal login dan keluar pesan error	User gagal login dan keluar pesan error
3	username = username value and password = password value	User sukses login dan tampilkan menu pemesanan	User sukses login dan tampilkan menu pemesanan

Sumber : [Pengujian]

6.1.2 Pengujian Proses Tambah Data

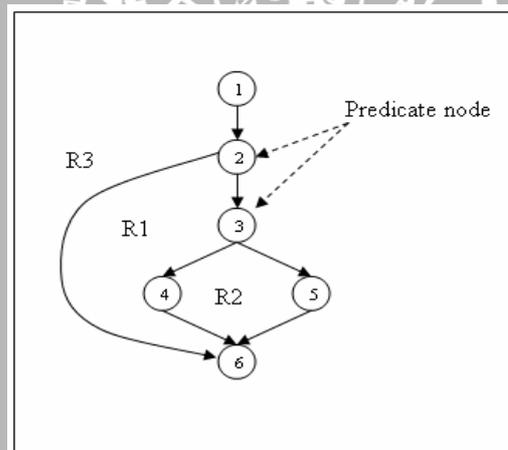
1. Pembuatan PDL untuk desain *test case* dengan simpel – simpel yang diidentifikasi



Gambar 6.3 PDL Proses Tambah Data

Sumber : [Pengujian]

2. Pembuatan grafik alir yang sesuai



Gambar 6.4 Flow Graph Proses Tambah Data

Sumber : [Pengujian]

3. Penentuan kompleksitas siklomatis dari grafik alir resultan

$$V(G) = 3 \text{ region}$$

$$V(G) = 7 \text{ edge} - 6 \text{ node} + 2 = 3$$

$$V(G) = 2 \text{ predicate node} + 1 = 3$$

4. Penentuan jalur independen

$$\text{Jalur 1 : } 1 - 2 - 6$$

$$\text{Jalur 2 : } 1 - 2 - 3 - 4 - 6$$

$$\text{Jalur 3 : } 1 - 2 - 3 - 5 - 6$$

5. Melakukan kasus uji

Penentuan kasus uji untuk masing-masing jalur dan hasil eksekusi untuk masing-masing kasus uji dijelaskan pada Tabel 6.2.

Tabel 6.2 Kasus uji untuk pengujian proses tambah data

Jalur	Kasus uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Data value = null	Data yang dimasukkan kosong	Data yang dimasukkan kosong
2	Id_data value != null and data value != null and query sql "gagal"	Data gagal ditambahkan	Data gagal ditambahkan
3	Id_data value != null and data value != null and query sql "sukses"	Data berhasil ditambahkan	Data berhasil ditambahkan

Sumber : [Pengujian]

6.1.3 Pengujian Proses Edit Data

1. Pembuatan PDL untuk desain *test case* dengan simpul – simpul yang diidentifikasi

```

PROCEDURE edit data

Prosedur untuk mengedit data
TYPE id, data IS STRING

1 {
CONNECT TO database
IF (GET(id))
SQL SELECT * FROM database WHERE id = GET(id) value
VIEW data
}

2 ← IF (data update found) THEN
3 {
SQL UPDATE FROM database WHERE id = id value
IF (update success) THEN
4 ← PRINT "Data Telah Diubah";
5 {
ELSE
PRINT "Data Gagal Diubah";
6 ← ENDF
7 ← ENDF
ENDIF

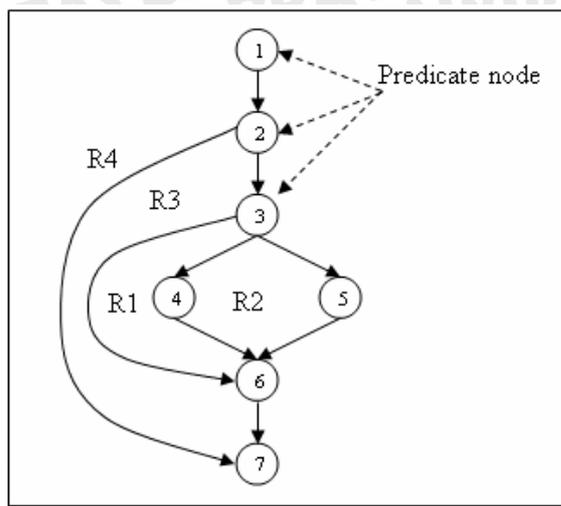
ENDPROC
    
```

Gambar 6.5 PDL Proses Edit Data

Sumber : [Pengujian]

2. Pembuatan grafik alir yang sesuai





Gambar 6.6 Flow Graph Proses Edit Data
Sumber : [Pengujian]

3. Penentuan kompleksitas siklomatis dari grafik alir resultan

$$V(G) = 4 \text{ region}$$

$$V(G) = 9 \text{ edge} - 7 \text{ node} + 2 = 4$$

$$V(G) = 3 \text{ predicate node} + 1 = 4$$

4. Penentuan jalur independen

Jalur 1 : 1 – 2 – 7

Jalur 2 : 1 – 2 – 3 – 6 – 7

Jalur 3 : 1 – 2 – 3 – 4 – 6 – 7

Jalur 4 : 1 – 2 – 3 – 5 – 6 – 7

5. Melakukan kasus uji

Penentuan kasus uji untuk masing-masing jalur dan hasil eksekusi untuk masing-masing kasus uji dijelaskan pada Tabel 6.3.

Tabel 6.3 Kasus uji untuk pengujian proses edit data

Jalur	Kasus uji	Hasil yang diharapkan	Hasil yang didapatkan
1	SQL update "gagal"	Data gagal diupdate	Data gagal diupdate
2	SQL update "sukses"	Data berhasil diupdate	Data berhasil diupdate

Sumber : [Pengujian]

6.1.4 Pengujian Proses Hapus Data

1. Pembuatan PDL untuk desain *test case* dengan simpul – simpul yang diidentifikasi

```

PROCEDURE hapus data

Prosedur untuk menghapus data
TYPE id IS STRING
CONNECT TO database

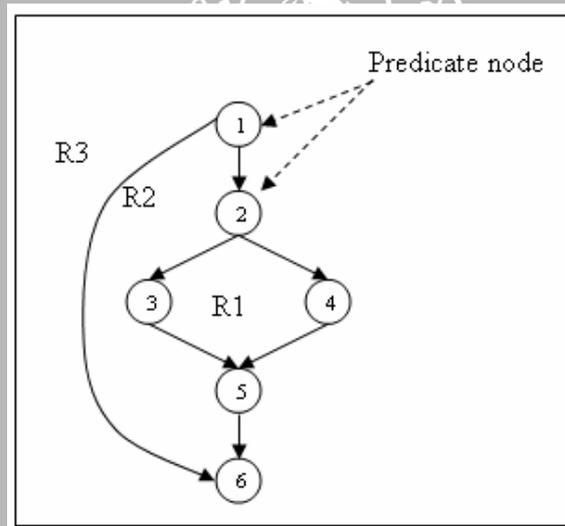
① IF (GET (id))
    SQL DELETE FROM database WHERE id = GET(id)
② ← IF (delete success) THEN
    ③ ← PRINT "Data Telah Dihapus";
    ④ ← ELSE
        PRINT "Data Gagal Dihapus";
    ⑤ ← ENDDIF
⑥ ← ENDDIF

ENDPROC
    
```

Gambar 6.7 PDL Proses Hapus Data

Sumber : [Pengujian]

2. Pembuatan grafik alir yang sesuai



Gambar 6.8 Flow Graph Proses Hapus Data

Sumber : [Pengujian]

3. Penentuan kompleksitas siklomatis dari grafik alir resultan

$$V(G) = 3 \text{ region}$$

$$V(G) = 7 \text{ edge} - 6 \text{ node} + 2 = 3$$

$$V(G) = 2 \text{ predicate node} + 1 = 3$$

4. Penentuan jalur independen

Jalur 1 : 1 – 6

Jalur 2 : 1 – 2 – 4 – 5 – 6

Jalur 3 : 1 – 2 – 3 – 5 – 6

5. Melakukan kasus uji

Penentuan kasus uji untuk masing-masing jalur dan hasil eksekusi untuk masing-masing kasus uji dijelaskan pada Tabel 6.4.

Table 6.4 Kasus uji untuk pengujian proses hapus data

Jalur	Kasus uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Id_data value = null	Obyek list dengan jumlah elemen 0	Obyek list dengan jumlah elemen 0
2	SQL delete "gagal"	Data gagal dihapus	Data gagal dihapus
3	SQL delete "sukses"	Data berhasil dihapus	Data berhasil dihapus

Sumber : [Pengujian]

6.2 Pengujian Validasi

Pengujian validasi digunakan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan. Item-item yang telah dirumuskan dalam daftar kebutuhan dan merupakan hasil analisis kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode pengujian *Black Box*, karena tidak memerlukan untuk berkonsentrasi terhadap alur jalannya algoritma program dan lebih ditekankan untuk menemukan konformitas antara kinerja sistem dengan daftar kebutuhan.

6.2.1 Kasus Uji Validasi Sistem WAP Pemesanan Jasa Travel

Untuk mengetahui kesesuaian antara kebutuhan dengan kinerja sistem WAP pada *handphone*, pada setiap kebutuhan (*requirement*) dilakukan proses pengujian dengan kasus uji masing-masing.

6.2.1.1 Kasus Uji Login

Kasus uji login ditunjukkan dalam Tabel 6.5a dan Tabel 6.5b.

Tabel 6.5a Kasus uji login dimana *username* dan *password* ada dalam sistem

Nama kasus uji	Kasus uji login
Obyek uji	Kebutuhan login ke sistem
Tujuan pengujian	Menguji bahwa hanya pengguna yang telah terdaftar dapat masuk dan mengakses fasilitas tertentu. Pengguna sistem yang tidak login hanya akan menjadi <i>user</i> pengunjung.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan. 2. Memasukkan <i>username</i> dan <i>password</i> pada form login (<i>username</i> = "solikin", <i>password</i> = "raharjo"). 3. Menekan tombol login.
Hasil yang diharapkan	Pengguna berhasil login pada sistem.

Sumber : [Pengujian]

Tabel 6.5b Kasus uji login dimana *username* dan *password* tidak ada dalam sistem

Nama kasus uji	Kasus uji login dimana <i>username</i> dan <i>password</i> tidak ada dalam sistem
Obyek uji	Kebutuhan login ke sistem
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan bahwa login tidak berhasil.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan. 2. Memasukkan <i>username</i> dan <i>password</i> pada form login (<i>username</i>="xxx", <i>password</i> = "xxx"). 3. Menekan tombol login.
Hasil yang diharapkan	Menampilkan pesan login tidak berhasil.

Sumber : [Pengujian]

6.2.1.2 Kasus Uji Logout

Kasus uji logout ditunjukkan dalam Tabel 6.6.

Tabel 6.6 Kasus uji logout

Nama kasus uji	Kasus uji logout
Obyek uji	Kebutuhan logout dari sistem
Tujuan pengujian	Menguji bahwa sistem dapat menyediakan fasilitas bagi <i>user</i> yang telah login untuk keluar dari sistem
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah <i>login</i> 2. <i>User</i> menekan tombol logout
Hasil yang diharapkan	<i>User</i> berhasil keluar dari sistem

Sumber : [Pengujian]

6.2.1.3 Kasus Uji Daftar Anggota Baru

Kasus uji daftar anggota baru yang dapat dilakukan oleh *user* pengunjung ditunjukkan dalam Tabel 6.7a, 6.7b, 6.7c

Tabel 6.7a Kasus Uji Daftar Anggota Baru

Nama kasus uji	Kasus Daftar Anggota Baru
Obyek uji	Kebutuhan Daftar Anggota
Tujuan pengujian	Menguji bahwa sistem dapat menyediakan fasilitas bagi <i>user</i> pengunjung yang ingin memesan jasa dengan cara menjadi anggota.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan 2. <i>User</i> menekan link "Registrasi Anggota" pada halaman utama. 3. <i>User</i> mengisi data yang diperlukan termasuk <i>username</i> dan <i>password</i>. 4. Setelah data diisi lengkap, <i>user</i> menekan tombol "enter" atau tekan link "option → daftar" 5. Tampil halaman selamat datang dan halaman menu anggota.

Hasil yang diharapkan	User pengunjung berhasil mendaftarkan diri menjadi anggota.
-----------------------	---

Sumber : [Pengujian]

Tabel 6.7b Kasus uji daftar anggota dimana masukan data kosong

Nama kasus uji	Kasus uji Daftar Anggota Baru dimana masukan data kosong
Obyek uji	Kebutuhan Daftar Anggota
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan bahwa <i>username</i> , <i>password</i> dan elemen lainnya tidak boleh kosong.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan 2. User menekan link “Registrasi Anggota” pada halaman utama. 3. Data yang diperlukan termasuk <i>username</i> dan <i>password</i> dikosongkan. 4. User menekan tombol “enter” atau tekan link “option → daftar”
Hasil yang diharapkan	Sistem menampilkan pesan kesalahan bahwa daftar user gagal dan data tidak boleh ada yang kosong

Sumber : [Pengujian]

Tabel 6.7c Kasus uji daftar anggota dimana *username* sudah ada

Nama kasus uji	Kasus uji Daftar Anggota Baru dimana masukan data <i>username</i> dan <i>password</i> sama dengan yang terdaftar di dalam sistem.
Obyek uji	Kebutuhan Daftar Anggota
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan bahwa <i>username</i> , <i>password</i> sudah terdaftar dalam sistem.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan 2. User menekan link “Registrasi Anggota” pada halaman utama. 3. Data yang diperlukan termasuk <i>username</i> dan <i>password</i> diisi sama dengan <i>user</i> yang sudah terdaftar. 4. User menekan tombol “enter” atau tekan link “option → daftar”
Hasil yang diharapkan	Sistem menampilkan pesan kesalahan bahwa <i>username</i> dan <i>password</i> sudah ada dalam sistem.

Sumber : [Pengujian]

6.2.1.3 Kasus Uji Pemesanan Jasa

Kasus uji pemesanan jasa yang dapat dilakukan oleh user yang telah mendaftar sebagai anggota ditunjukkan dalam tabel 6.8a, 6.8b

Tabel 6.8a Kasus uji pemesanan jasa dan dikonfirmasi oleh anggota

Nama kasus uji	Kasus uji Pemesanan Jasa oleh anggota.
Obyek uji	Kebutuhan Pemesanan Jasa
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan menu pemesanan jasa dan user dapat melakukan pemesanan jasa.
Prosedur uji	<ol style="list-style-type: none">1. Sistem telah berjalan dan <i>user</i> telah login.2. <i>User</i> menekan link “Jadwal Keberangkatan” pada halaman anggota.3. <i>User</i> menekan link “pesan travel”.4. <i>User</i> memilih tanggal keberangkatan, jam berangkat dan jumlah penumpang.5. <i>User</i> memasukkan alamat jemput dan tujuan.6. Tampil Halaman konfirmasi pemesanan jasa.7. <i>User</i> menekan tombol “setuju”.8. Proses pemesanan berhasil beserta tampilan kode pemesanan.
Hasil yang diharapkan	Sistem menampilkan halaman keberhasilan pemesanan beserta kode pemesanan.

Sumber : [Pengujian]

Tabel 6.8b Kasus uji pemesanan jasa yang tidak dikonfirmasi oleh anggota

Nama kasus uji	Kasus uji Pemesanan Jasa oleh anggota tanpa konfirmasi.
Obyek uji	Kebutuhan Pemesanan Jasa
Tujuan pengujian	Menguji bahwa sistem hanya dapat menampilkan data pemesanan yang telah dikonfirmasi oleh <i>user</i> anggota terlebih dahulu.
Prosedur uji	<ol style="list-style-type: none">1. Sistem telah berjalan dan <i>user</i> telah login.2. <i>User</i> menekan link “Jadwal Keberangkatan” pada halaman anggota.3. <i>User</i> menekan link “pesan travel”.4. <i>User</i> memilih tanggal keberangkatan, jam berangkat dan jumlah penumpang.5. <i>User</i> memasukkan alamat jemput dan tujuan.6. Tampil Halaman konfirmasi pemesanan jasa.7. <i>User</i> menekan tombol “tidak setuju”.8. Proses pemesanan tidak berhasil dan kembali ke halaman “Jadwal Keberangkatan”.
Hasil yang diharapkan	Sistem tidak menyimpan data pemesanan anggota karena tidak adanya konfirmasi pemesanan dari anggota.

Sumber : [Pengujian]

6.2.1.4 Kasus Uji Pembatalan Pemesanan

Kasus uji pembatalan pemesanan jasa yang dapat dilakukan oleh *user* yang telah mendaftar sebagai anggota ditunjukkan dalam tabel 6.9

Tabel 6.9 Kasus uji pembatalan pemesanan jasa oleh anggota

Nama kasus uji	Kasus uji Pembatalan Pemesanan Jasa oleh anggota.
Obyek uji	Kebutuhan Pembatalan Pemesanan Jasa
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan menu pembatalan pemesanan jasa dan user dapat melakukan pembatalan pemesanan jasa.
Prosedur uji	<ol style="list-style-type: none">1. Sistem telah berjalan dan <i>user</i> telah login.2. <i>User</i> menekan link “Pembatalan Pemesanan” pada halaman anggota.3. <i>User</i> mengisi kode pemesanan yang telah diberitahukan sebelumnya pada saat konfirmasi pemesanan.4. <i>User</i> memasukkan alamat jemput dan tujuan.5. Pemesanan berhasil dibatalkan
Hasil yang diharapkan	Sistem berhasil menghapus data pemesanan <i>user</i> berdasarkan kode pemesanan.

Sumber : [Pengujian]

6.2.1.5 Kasus Uji Lihat Info Pemesanan Anggota

Kasus uji lihat info pemesanan anggota yang dapat dilakukan oleh *user* yang telah terdaftar sebagai anggota dan telah melakukan pemesanan jasa ditunjukkan dalam tabel 6.10a, 6.10b.

Tabel 6.10 Kasus Uji Lihat Info Pemesanan Anggota

Nama kasus uji	Kasus uji Lihat Info Pemesanan anggota.
Obyek uji	Kebutuhan melihat info pemesanan dari anggota
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan menu info pemesanan jasa.
Prosedur uji	<ol style="list-style-type: none">1. Sistem telah berjalan dan <i>user</i> telah login.2. <i>User</i> menekan link “Lihat Pesanan” pada halaman anggota.3. Halaman info transaksi pemesanan jasa tampil.
Hasil yang diharapkan	Sistem berhasil menampilkan halaman info transaksi pemesanan.

Sumber : [Pengujian]

Tabel 6.10b Kasus Uji Lihat Info Pemesanan Anggota tanpa adanya data pemesanan.

Nama kasus uji	Kasus uji Lihat Info Pemesanan anggota tanpa ada data pemesanan.
Obyek uji	Kebutuhan melihat info pemesanan dari anggota
Tujuan pengujian	Menguji bahwa sistem tidak dapat menampilkan menu info pemesanan jasa tanpa memesan terlebih dahulu.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login. 2. <i>User</i> menekan link “Lihat Pesanan” pada halaman anggota. 3. Halaman info transaksi pemesanan jasa tampil dengan isi halaman kosong (tidak ada data pemesanan yang ditampilkan).
Hasil yang diharapkan	Sistem menampilkan halaman info transaksi pemesanan dengan isi halaman kosong (tanpa data).

Sumber : [Pengujian]

6.2.2 Kasus Uji Validasi Sistem WEB Pemesanan Jasa Travel

Untuk mengetahui kesesuaian antara kebutuhan dengan kinerja sistem WEB pada komputer, pada setiap kebutuhan (*requirement*) dilakukan proses pengujian dengan kasus uji masing-masing.

6.2.2.1 Kasus Uji Login *Customer Service*

Kasus uji validasi login *customer service* yang hanya dapat dilakukan oleh admin ditunjukkan dalam tabel 6.11a dan 6.11b

Tabel 6.11 Kasus Uji Validasi Login *Customer Service*

Nama kasus uji	Kasus uji Validasi Login <i>Customer Service</i>
Obyek uji	Kebutuhan login ke sistem sebagai <i>customer service</i> .
Tujuan pengujian	Menguji bahwa hanya pengguna yang telah terdaftar dapat masuk dan mengakses fasilitas tertentu. Pengguna sistem yang tidak login sebagai <i>customer service</i> maka tidak dapat mengakses fasilitas <i>customer service</i> .
Prosedur uji	<ol style="list-style-type: none"> 1. <i>Customer Service</i> memberikan <i>username</i> dan <i>passwordnya</i> untuk dimasukkan ke dalam sistem oleh admin. 2. Sistem telah berjalan dan login sebagai <i>customer service</i>. 3. Halaman <i>input</i> data baru dan <i>edit</i> data yang telah ada dari sistem pemesanan jasa travel tampil.
Hasil yang diharapkan	Sistem menampilkan halaman edit dan input data yang hanya dapat diakses jika login sebagai <i>customer service</i> .

Sumber : [Pengujian]

6.2.2.2 Kasus Uji Login Manajemen

Kasus uji validasi login manajemen yang dapat dilakukan oleh admin ditunjukkan dalam tabel 6.12

Tabel 6.12 Kasus Uji Validasi Login Manajemen

Nama kasus uji	Kasus uji Validasi Login Manajemen
Obyek uji	Kebutuhan login ke sistem sebagai manajemen.
Tujuan pengujian	Menguji bahwa hanya pengguna yang telah terdaftar dapat masuk dan mengakses fasilitas tertentu. Pengguna sistem yang tidak login sebagai manajemen maka tidak dapat mengakses laporan sistem pemesanan jasa.
Prosedur uji	<ol style="list-style-type: none"> 1. Pihak manajemen memberikan <i>username</i> dan <i>passwordnya</i> untuk dimasukkan ke dalam sistem oleh admin. 2. Sistem telah berjalan dan login sebagai manajemen. 3. Halaman lihat laporan baik laporan pemesanan, pembatalan, pembayaran dan data anggota dari sistem pemesanan jasa travel tampil.
Hasil yang diharapkan	Sistem menampilkan halaman lihat laporan sistem pemesanan jasa yang hanya dapat diakses jika login sebagai manajemen.

Sumber : [Pengujian]

6.2.2.3 Kasus Uji Edit Data Sistem Jasa Travel

Kasus uji edit data sistem pemesanan jasa travel yang dilakukan oleh *customer service* ditunjukkan dalam tabel 6.13a dan 6.13b

Tabel 6.13a Kasus Uji Edit Data sistem jasa travel

Nama kasus uji	Kasus uji Edit Data Sistem Jasa Travel
Obyek uji	Kebutuhan edit data travel
Tujuan pengujian	Menguji fungsi edit data oleh <i>customer service</i> pada sistem pemesanan jasa travel.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem berjalan dan login sebagai <i>customer service</i> 2. Tampil halaman edit data . 3. <i>Customer Service</i> melakukan editing pada data <i>kota_tujuan</i> (mengganti kota yang dituju beserta kode kota). 4. Menyimpan hasil edit data tersebut.
Hasil yang diharapkan	Sistem dapat melakukan proses edit data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> .

Sumber : [Pengujian]

Tabel 6.13b Kasus Uji Edit Data sistem jasa travel tanpa melengkapi *form* isian.

Nama kasus uji	Kasus uji Edit Data Sistem Jasa Travel tanpa melengkapi <i>form</i> isian
Obyek uji	Kebutuhan edit data travel
Tujuan pengujian	Menguji bahwa sistem tidak dapat menyimpan data jika ada <i>form edit data</i> yang belum diisi.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem berjalan dan login sebagai <i>customer service</i> 2. Tampil halaman edit data . 3. <i>Customer Service</i> melakukan editing pada data <i>kota_tujuan</i> dengan mengganti kota yang dituju tanpa mengisi kode kota). 4. Menyimpan hasil edit data tersebut. 5. Tampil pesan bahwa ada kesalahan dalam pengeditan data karena ada <i>form</i> yang belum diisi.
Hasil yang diharapkan	Sistem tidak dapat melakukan proses edit data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> karena data yang diisikan tidak lengkap.

Sumber : [Pengujian]

6.2.2.4 Kasus Uji Melakukan *Update* Data

Kasus uji melakukan *update* data yang dilakukan oleh *customer service* ditunjukkan dalam tabel 6.14a dan 6.14b.

Tabel 6.14a Kasus Uji *Update* Data sistem jasa travel

Nama kasus uji	Kasus uji <i>Update</i> Data Sistem Jasa Travel
Obyek uji	Kebutuhan memasukkan data baru pada sistem
Tujuan pengujian	Menguji fungsi <i>update</i> data oleh <i>customer service</i> pada sistem pemesanan jasa travel.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem berjalan dan login sebagai <i>customer service</i> 2. Tampil halaman <i>input</i> data . 3. <i>Customer Service</i> memasukkan data kota tujuan yang baru pada <i>database kota_tujuan</i>. 4. Menyimpan hasil <i>update</i> data tersebut.
Hasil yang diharapkan	Sistem dapat melakukan proses <i>update</i> data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> .

Sumber : [Pengujian]

Tabel 6.14b Kasus Uji *Update* Data sistem jasa travel tanpa mengisi *form* isian.

Nama kasus uji	Kasus uji <i>Update</i> Data Sistem Jasa Travel tanpa mengisi <i>form</i> isian.
Obyek uji	Kebutuhan memasukkan data baru pada sistem
Tujuan pengujian	Menguji bahwa sistem tidak dapat menyimpan data yang baru jika ada <i>form input data</i> yang belum diisi.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem berjalan dan login sebagai <i>customer service</i>

	<ol style="list-style-type: none"> 2. Tampil halaman <i>input</i> data . 3. <i>Customer Service</i> mengosongkan <i>form</i> isian pada halaman <i>input</i> data <i>kota_tujuan</i>. 4. Menyimpan hasil <i>update</i> data tersebut. 5. Tampil pesan kesalahan bahwa <i>input</i> data kosong.
Hasil yang diharapkan	Sistem tidak dapat melakukan proses <i>update</i> data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> karena ada <i>form</i> isian yang kosong.

Sumber : [Pengujian]

6.2.2.5 Kasus Uji Hapus Data

Kasus uji hapus data ini dilakukan oleh *customer service* ditunjukkan dalam tabel 6.15.

Tabel 6.15 Kasus Uji Hapus Data sistem jasa travel.

Nama kasus uji	Kasus uji Hapus Data Sistem Jasa Travel
Obyek uji	Kebutuhan menghapus data pada sistem
Tujuan pengujian	Menguji bahwa sistem dapat menghapus data yang diinginkan.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem berjalan dan login sebagai <i>customer service</i> 2. Tampil halaman hapus data . 3. <i>Customer Service</i> menghapus data sistem travel yang diinginkan . 4. Menyimpan hasil <i>update</i> data tersebut. 5. Tampil pesan bahwa data sudah dihapus.
Hasil yang diharapkan	Sistem dapat melakukan proses hapus data, dan menyimpan perubahan tersebut ke dalam <i>database</i> .

Sumber : [Pengujian]

6.2.2.6 Kasus Uji Lihat Laporan

Kasus uji lihat laporan yang dilakukan oleh *user* manajemen ditunjukkan dalam tabel 6.16

Tabel 6.16 Kasus Uji Lihat Laporan

Nama kasus uji	Kasus uji lihat laporan Sistem Jasa Travel
Obyek uji	Kebutuhan menampilkan laporan sistem pemesanan jasa travel.
Tujuan pengujian	Menguji fungsi lihat laporan bagi <i>user</i> manajemen pada sistem pemesanan jasa travel.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem berjalan dan login sebagai <i>user</i> manajemen. 2. Tampil halaman lihat laporan . 3. <i>User</i> manajemen memilih laporan apa saja yang ingin di tampilkan 4. Menampilkan laporan yang diinginkan oleh <i>user</i>

	manajemen.
Hasil yang diharapkan	Sistem dapat melakukan proses menampilkan laporan sistem pemesanan jasa travel sesuai permintaan

Sumber : [Pengujian]

6.2.2.7 Kasus Uji Lihat Daftar Anggota

Kasus uji melihat daftar user yang hanya bisa dilakukan oleh admin ditunjukkan dalam Tabel 6.17.

Tabel 6.17 Kasus uji melihat daftar user anggota

Nama kasus uji	Kasus uji melihat daftar user anggota
Obyek uji	Kebutuhan menampilkan daftar user anggota
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan semua user anggota yang tercatat di dalam sistem.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu “anggota”. 3. Tampil halaman daftar anggota.
Hasil yang diharapkan	Semua daftar user yang telah terdaftar dapat ditampilkan.

Sumber : [Pengujian]

6.2.2.8 Kasus Uji *Edit Data User*

Kasus uji melakukan *edit* terhadap data anggota yang telah terdaftar hanya bisa dilakukan oleh admin ditunjukkan dalam tabel 6.18

Tabel 6.18a Kasus uji mengedit data user anggota

Nama kasus uji	Kasus uji mengedit data user anggota
Obyek uji	Kebutuhan mengedit data user anggota
Tujuan pengujian	Menguji bahwa sistem dapat mengedit data user anggota yang telah terdaftar.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu edit user. 3. Memilih <i>username</i> yang akan di edit. 4. Memasukkan data yang diperlukan pada <i>form</i> isian termasuk dapat mengubah <i>username</i> dan <i>password</i> anggota. 5. Menekan tombol simpan.
Hasil yang diharapkan	Data user termasuk <i>username</i> dan <i>password</i> berhasil diedit dan disimpan.

Sumber : [Pengujian]

Tabel 6.18b Kasus uji mengedit data user *customer service*

Nama kasus uji	Kasus uji mengedit data user <i>customer service</i>
Obyek uji	Kebutuhan mengedit data user <i>customer service</i>
Tujuan pengujian	Menguji bahwa sistem dapat mengedit data user <i>customer service</i> yang telah terdaftar.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu edit user <i>customer service</i>. 3. Memilih <i>username</i> yang akan di edit. 4. Memasukkan data <i>username</i> dan <i>password</i> yang baru pada <i>form</i> isian. 5. Menekan tombol simpan.
Hasil yang diharapkan	Data user <i>customer service</i> berhasil diedit dan disimpan.

Sumber : [Pengujian]

Tabel 6.18c Kasus uji mengedit data user manajemen

Nama kasus uji	Kasus uji mengedit data user manajemen
Obyek uji	Kebutuhan mengedit data user manajemen
Tujuan pengujian	Menguji bahwa sistem dapat mengedit data user manajemen yang telah terdaftar.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu edit user manajemen. 3. Memilih <i>username</i> yang akan di edit. 4. Memasukkan data <i>username</i> dan <i>password</i> yang baru pada <i>form</i> isian. 5. Menekan tombol simpan.
Hasil yang diharapkan	Data user manajemen berhasil diedit dan disimpan.

Sumber : [Pengujian]

Tabel 6.18d Kasus uji mengedit data user dimana *username* tidak boleh kosong

Nama kasus uji	Kasus uji mengedit data dimana <i>username</i> tidak boleh kosong
Obyek uji	Kebutuhan mengedit data user
Tujuan pengujian	Menguji bahwa sistem dapat menampilkan pesan kesalahan bahwa data <i>username</i> tidak boleh kosong.
Prosedur uji	<ol style="list-style-type: none"> 1. Sistem telah berjalan dan <i>user</i> telah login sebagai Admin. 2. Memilih menu edit dan hapus user. 3. Memasukkan data pada <i>form</i> (<i>username</i>=" ") 4. Menekan tombol simpan
Hasil yang diharapkan	Menampilkan pesan kesalahan gagal edit user

Sumber : [Pengujian]

6.2.2.9 Kasus Uji Hapus Data User

Kasus uji menghapus user yang hanya bisa dilakukan oleh admin ditunjukkan dalam Tabel 6.19.

Tabel 6.19 Kasus uji menghapus data user

Nama kasus uji	Kasus uji menghapus data user
Obyek uji	Kebutuhan menghapus data user
Tujuan pengujian	Menguji bahwa sistem dapat menghapus data user
Prosedur uji	1. Sistem telah berjalan dan <i>user</i> login sebagai Admin. 2. Memilih menu edit dan hapus user. 3. Memilih user yang akan dihapus dan menekan tombol “hapus”.
Hasil yang diharapkan	User bisa dihapus dari sistem

Sumber : [Pengujian]

6.2.3 Hasil Pengujian Validasi

Dari kasus uji yang telah dilaksanakan pada sistem WAP dan sistem WEB Pemesanan Jasa Travel sesuai dengan prosedur pengujian pada sub pokok bahasan 6.2.1 dan 6.2.2, didapatkan hasil seperti ditunjukkan pada Tabel 6.20 dan 6.21.

Tabel 6.20 Hasil Pengujian Validasi Sistem WAP Pemesanan Jasa Travel

No	Nama kasus uji	Hasil yang diharapkan	Hasil yang didapat	Status
1	Kasus uji <i>login</i>	Pengguna berhasil login pada sistem.	Pengguna berhasil login pada sistem.	Valid
2	Kasus uji login dimana <i>username</i> dan <i>password</i> tidak ada dalam sistem	Menampilkan pesan login tidak berhasil.	Menampilkan pesan login tidak berhasil.	Valid
3	Kasus uji <i>logout</i>	Pengguna berhasil keluar dari sistem.	Pengguna berhasil keluar dari sistem.	Valid
4	Kasus uji daftar anggota baru	User pengunjung berhasil mendaftarkan diri menjadi anggota.	User pengunjung berhasil mendaftarkan diri menjadi anggota.	Valid
5	Kasus uji daftar anggota baru dimana masukan data kosong	Sistem menampilkan pesan kesalahan bahwa daftar user gagal dan data tidak boleh ada yang kosong	Sistem menampilkan pesan kesalahan bahwa daftar user gagal dan data tidak boleh ada yang kosong	Valid
6	Kasus uji daftar anggota baru dimana <i>username</i> sudah ada	Sistem menampilkan pesan kesalahan bahwa <i>username</i> dan <i>password</i> sudah ada dalam sistem.	Sistem menampilkan pesan kesalahan bahwa <i>username</i> dan <i>password</i> sudah ada dalam sistem.	Valid
7	Kasus uji Pemesanan Jasa	Sistem menampilkan halaman keberhasilan pemesanan beserta kode pemesanan.	Sistem menampilkan halaman keberhasilan pemesanan beserta kode pemesanan.	Valid
8	Kasus uji Pemesanan Jasa yang tidak dikonfirmasi oleh anggota	Sistem tidak menyimpan data pemesanan anggota karena tidak adanya konfirmasi pemesanan dari anggota.	Sistem tidak menyimpan data pemesanan anggota karena tidak adanya konfirmasi pemesanan dari anggota.	Valid
9	Kasus uji Pembatalan Pemesanan Jasa	Sistem berhasil menghapus data pemesanan <i>user</i>	Sistem berhasil menghapus data pemesanan <i>user</i>	Valid

				berdasarkan kode pemesanan.	berdasarkan kode pemesanan.	
10	Kasus uji Lihat Info Transaksi Anggota Pemesanan			Sistem berhasil menampilkan halaman info transaksi pemesanan berdasarkan kode transaksi.	Sistem berhasil menampilkan halaman info transaksi pemesanan berdasarkan kode transaksi.	Valid
11	Kasus uji Lihat Info Transaksi Anggota Pemesanan tanpa adanya data transaksi pemesanan.			Sistem menampilkan halaman info transaksi pemesanan dengan isi halaman kosong (tanpa data).	Sistem menampilkan halaman info transaksi pemesanan dengan isi halaman kosong (tanpa data).	Valid

Sumber : [Pengujian]

Tabel 6.21 Hasil Pengujian Validasi Sistem WEB Pemesanan Jasa Travel

No	Nama kasus uji	Hasil yang diharapkan	Hasil yang didapat	Status
1	Kasus uji <i>login customer service</i>	Sistem menampilkan halaman edit dan input data yang hanya dapat diakses jika login sebagai <i>customer service</i> .	Sistem menampilkan halaman edit dan input data yang hanya dapat diakses jika login sebagai <i>customer service</i> .	Valid
2	Kasus uji login manajemen	Sistem menampilkan halaman lihat laporan sistem pemesanan jasa yang hanya dapat diakses jika login sebagai manajemen.	Sistem menampilkan halaman lihat laporan sistem pemesanan jasa yang hanya dapat diakses jika login sebagai manajemen.	Valid
3	Kasus uji login admin	Sistem menampilkan halaman edit data user yang hanya dapat diakses jika login sebagai admin.	Sistem menampilkan halaman edit data user yang hanya dapat diakses jika login sebagai admin.	Valid
4	Kasus uji edit data sistem jasa travel	Sistem dapat melakukan proses edit data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> .	Sistem dapat melakukan proses edit data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> .	Valid
5	Kasus uji Edit Data Sistem Jasa Travel tanpa melengkapi <i>form</i> isian.	Sistem tidak dapat melakukan proses edit data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> karena data yang diisikan tidak lengkap.	Sistem tidak dapat melakukan proses edit data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> karena data yang diisikan tidak lengkap.	Valid
6	Kasus uji <i>Update</i> Data Sistem Jasa Travel	Sistem dapat melakukan proses <i>update</i> data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> .	Sistem dapat melakukan proses <i>update</i> data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> .	Valid
7	Kasus uji <i>Update</i> Data Sistem Jasa Travel tanpa mengisi <i>form</i> isian.	Sistem tidak dapat melakukan proses <i>update</i> data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> karena ada <i>form</i> isian yang kosong.	Sistem tidak dapat melakukan proses <i>update</i> data, dalam hal ini data <i>kota_tujuan</i> dan menyimpan perubahan tersebut ke dalam <i>database</i> karena ada <i>form</i> isian yang kosong.	Valid
8	Kasus uji lihat laporan Sistem Jasa Travel	Sistem dapat melakukan proses menampilkan laporan sistem pemesanan jasa	Sistem dapat melakukan proses menampilkan laporan sistem pemesanan jasa	Valid



		travel sesuai permintaan <i>user</i> manajemen.	travel sesuai permintaan <i>user</i> manajemen.	
9	Kasus uji melihat daftar user anggota	Semua daftar user yang telah terdaftar dapat ditampilkan.	Semua daftar user yang telah terdaftar dapat ditampilkan.	Valid
10	Kasus uji mengedit data user anggota	Data user termasuk <i>username</i> dan <i>password</i> berhasil diedit dan disimpan.	Data user termasuk <i>username</i> dan <i>password</i> berhasil diedit dan disimpan.	Valid
11	Kasus uji mengedit data user <i>customer service</i>	Data user <i>customer service</i> berhasil diedit dan disimpan.	Data user <i>customer service</i> berhasil diedit dan disimpan.	Valid
12	Kasus uji mengedit data user manajemen	Data user manajemen berhasil diedit dan disimpan.	Data user manajemen berhasil diedit dan disimpan.	Valid
13	Kasus uji mengedit data dimana <i>username</i> tidak boleh kosong	Menampilkan pesan kesalahan gagal edit user	Menampilkan pesan kesalahan gagal edit user	Valid
14	Kasus uji menghapus data user	User bisa dihapus dari sistem	User bisa dihapus dari sistem	Valid
15	Kasus uji Hapus Data Sistem Pemesanan Jasa Travel	Sistem dapat melakukan proses hapus data, dan menyimpan perubahan tersebut ke dalam <i>database</i> .	Sistem dapat melakukan proses hapus data, dan menyimpan perubahan tersebut ke dalam <i>database</i> .	Valid

Sumber : [Pengujian]

Dari 26 kasus uji yang ditunjukkan dalam Tabel 6.20 dan Tabel 6.21 dapat disimpulkan bahwa sistem yang dibangun tidak terdapat kesalahan dan terjamin validitasnya.

6.3 Pengujian Kecepatan Query

a. Tujuan

Pengujian dilakukan untuk mengetahui waktu yang dibutuhkan untuk melakukan sebuah *query* pada *database* Sistem Pemesanan Jasa Travel Berbasis WAP. Pengujian kecepatan *query* ini menggunakan contoh *query* transaksi pemesanan jasa yang dilakukan oleh *user* anggota berdasarkan tabel pesan.

Pengujian dilakukan untuk mendapatkan perbandingan waktu *query* yang dilakukan terhadap jumlah data yang berbeda pada *database* Sistem Pemesanan Jasa Travel Berbasis WAP.

b. Prosedur Pengujian

- Mengakses test_query.php. Halaman ini dibuat khusus untuk melakukan pengujian waktu akses *query*.

- Tabel pesan diberi masukan data sebanyak 100, 300, 500, 700, 900, dan 1100 *data entry* dalam *database* Sistem Pemesanan Jasa Travel Berbasis WAP.
- Melakukan perhitungan untuk mendapatkan nilai waktu akses *query* yang dilakukan terhadap jumlah data yang berbeda dalam *database* Sistem Pemesanan Jasa Travel.

c. Hasil Pengujian dan Analisis

- Hasil pengujian waktu *query* transaksi pemesanan jasa dengan jumlah data table pesan sebanyak 100, 300, 500, 700, 900, dan 1100 *data entry* ditunjukkan dalam Gambar 6.8.

<p>Hasil Query <i>select count(*) from pesan_copy</i></p> <p>Loop : 5 Jumlah Data : 100 record 100 data</p> <hr/> <p>Loop ke : 1 Waktu Akses : 0.000485897064209 detik Loop ke : 2 Waktu Akses : 0.000328063964844 detik Loop ke : 3 Waktu Akses : 0.000319004058838 detik Loop ke : 4 Waktu Akses : 0.000289916992188 detik Loop ke : 5 Waktu Akses : 0.00028395652771 detik</p> <hr/> <p>Total rata-rata waktu akses = 0.000341367721558 detik</p>	<p>Hasil Query <i>select count(*) from pesan_copy</i></p> <p>Loop : 5 Jumlah Data : 300 record 300 data</p> <hr/> <p>Loop ke : 1 Waktu Akses : 0.000650882720947 detik Loop ke : 2 Waktu Akses : 0.000335931777954 detik Loop ke : 3 Waktu Akses : 0.000308036804199 detik Loop ke : 4 Waktu Akses : 0.000275135040283 detik Loop ke : 5 Waktu Akses : 0.000272035598755 detik</p> <hr/> <p>Total rata-rata waktu akses = 0.000368404388428 detik</p>
<p>Hasil Query <i>select count(*) from pesan_copy</i></p> <p>Loop : 5 Jumlah Data : 500 record 500 data</p> <hr/> <p>Loop ke : 1 Waktu Akses : 0.000397920608521 detik Loop ke : 2 Waktu Akses : 0.000336170196533 detik Loop ke : 3 Waktu Akses : 0.000581979751587 detik Loop ke : 4 Waktu Akses : 0.000325202941895 detik Loop ke : 5 Waktu Akses : 0.000304937362671 detik</p> <hr/> <p>Total rata-rata waktu akses = 0.000389242172241 detik</p>	<p>Hasil Query <i>select count(*) from pesan_copy</i></p> <p>Loop : 5 Jumlah Data : 700 record 700 data</p> <hr/> <p>Loop ke : 1 Waktu Akses : 0.000381946563721 detik Loop ke : 2 Waktu Akses : 0.000350952148438 detik Loop ke : 3 Waktu Akses : 0.000296115875244 detik Loop ke : 4 Waktu Akses : 0.000284910202026 detik Loop ke : 5 Waktu Akses : 0.000717878341675 detik</p> <hr/> <p>Total rata-rata waktu akses = 0.000406360626221 detik</p>
<p>Hasil Query <i>select count(*) from pesan_copy</i></p> <p>Loop : 5 Jumlah Data : 900 record 900 data</p> <hr/> <p>Loop ke : 1 Waktu Akses : 0.000385999679565 detik Loop ke : 2 Waktu Akses : 0.000452041625977 detik Loop ke : 3 Waktu Akses : 0.00032114982605 detik Loop ke : 4 Waktu Akses : 0.000302076339722 detik Loop ke : 5 Waktu Akses : 0.000651836395264 detik</p> <hr/> <p>Total rata-rata waktu akses = 0.000422620773315 detik</p>	<p>Hasil Query <i>select count(*) from pesan_copy</i></p> <p>Loop : 5 Jumlah Data : 1100 record 1100 data</p> <hr/> <p>Loop ke : 1 Waktu Akses : 0.000499963760376 detik Loop ke : 2 Waktu Akses : 0.000332832336426 detik Loop ke : 3 Waktu Akses : 0.000339031219482 detik Loop ke : 4 Waktu Akses : 0.000741004943848 detik Loop ke : 5 Waktu Akses : 0.000324964523315 detik</p> <hr/> <p>Total rata-rata waktu akses = 0.000447559356689 detik</p>

Gambar 6.8 Hasil Pengujian Akses *Query*

Sumber : *Pengujian*

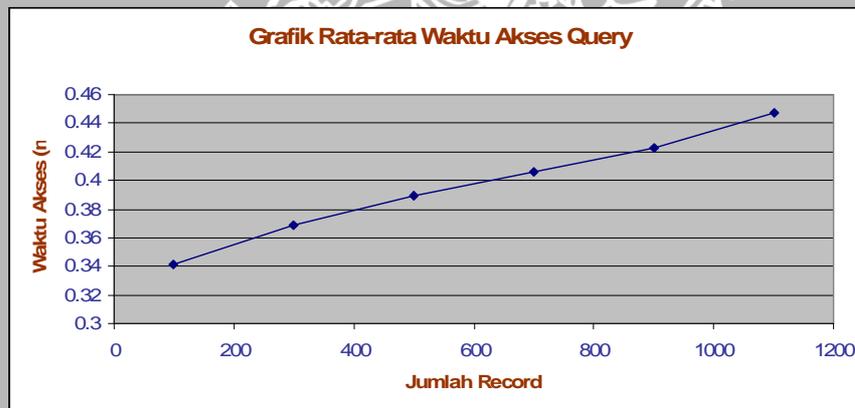
- Rata-rata pengujian waktu akses *query* transaksi pemesanan dengan jumlah data Tabel pesan sebanyak 100, 300, 500, 700, 900, dan 1100 data *entry* ditunjukkan dalam Tabel 6.23.

Tabel 6.23 Tabel Pengujian Waktu Akses *Query*

Pengujian	Jumlah <i>record</i>	Waktu akses (ms)
	Tabel pesan	
1	100	0.341368
2	300	0.368404
3	500	0.389242
4	700	0.406361
5	900	0.422621
6	1100	0.447559

Sumber : *Pengujian*

- Grafik perbandingan waktu akses *query* pada transaksi pemesanan dengan jumlah data yang berbeda diperlihatkan di dalam Gambar 6.9.



Gambar 6.9 Grafik Rata-Rata Waktu Akses *Query*

Sumber : *Pengujian*

- Dari pengujian waktu akses *query* dapat diketahui waktu akses per *record* untuk *query* transaksi pemesanan, merupakan hasil bagi antara waktu akses dengan jumlah *record*. Perbandingan waktu akses per *record* dari *query* transaksi pemesanan ditunjukkan dalam Tabel 6.24.

Tabel 6.24 Tabel Perbandingan Waktu Akses Per *Record*

Pengujian	Jumlah <i>record</i>	Waktu akses (ms)	Waktu akses per <i>record</i> (ms)
1	100	0.341368	0.00341368
2	300	0.368404	0.001228013
3	500	0.389242	0.000778484
4	700	0.406361	0.000580516
5	900	0.422621	0.000469579
6	1100	0.447559	0.000406872
Rata-rata waktu akses query per <i>record</i> (ms)			0.001146191

Sumber : *Pengujian*

Berdasarkan hasil pengujian pada Tabel 6.24 waktu akses *query* untuk *query* transaksi pemesanan dengan jumlah data Tabel pesan sebanyak 100, 300, 500, 700, 900, dan 1100 data *entry*, dapat diketahui bahwa rata-rata waktu akses per *record* sebesar 0,001146191 ms. Dari hasil pengujian dan analisis dapat diketahui bahwa waktu akses rata-rata *query* transaksi pemesanan dengan jumlah data yang ditampilkan semakin banyak, maka waktu akses akan meningkat secara linear.

BAB VII

PENUTUP

7.1 Kesimpulan

Berdasarkan hasil pengembangan sistem yang telah dilakukan, maka diambil kesimpulan sebagai berikut :

1. Sistem pemesanan dengan layanan WAP ini berhasil dikembangkan dengan bahasa pemrograman PHP dan WML dengan MySQL sebagai databasenya.
2. Pengujian algoritma dan pengujian validasi yang dilakukan telah menunjukkan bahwa sistem yang dibangun adalah valid sesuai dengan kebutuhan user.
3. Dengan adanya *handphone* yang sudah mendukung layanan WAP dan terdapat jaringan GPRS, maka Sistem Pemesanan Jasa Travel ini sudah dapat diakses.
4. Pengujian algoritma dan pengujian validasi yang dilakukan telah menunjukkan bahwa sistem yang dibangun adalah valid sesuai dengan kebutuhan user.

7.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem ini antara lain :

1. Sistem Pemesanan dengan layanan WAP ini dapat digunakan sebagai contoh dalam membangun sistem-sistem untuk semua jenis jasa pemesanan.
2. Untuk Sistem Antar Barang dapat diperinci lagi tentang batasan-batasan yang harus ada dalam masalah antar barang, contoh : berat barang, jumlah barang yang dapat dikirim perhari, dsb.
3. Untuk Informasi Transaksi Pemesanan yang sudah dilakukan oleh anggota dapat dikirimkan melalui SMS tanpa harus login ke dalam sistem pemesanan melalui WAP untuk melihat informasinya.
4. Sistem ini dapat dikembangkan lebih lanjut dengan menggunakan teknologi WAP 2.0 yang mempunyai fitur-fitur yang lebih lengkap dan tampilan yang lebih menarik.
5. Sistem ini dapat dikembangkan agar dapat memiliki tingkat keamanan yang lebih baik dari sisi perangkat lunak. Keamanan yang dimaksud seperti penggunaan algoritma enkripsi untuk *login*, anti sql injektion, penambahan hak otorisasi pada pengguna, dan lain sebagainya.

DAFTAR PUSTAKA

- [ANI-09] Anonymous, 2009, "Pengenalan PHP".
[http://www.deptan.go.id/pusdatin/admin/RB/Programming/Materi%20HP.pdf](http://www.deptan.go.id/pusdatin/admin/RB/Programming/Materi%20PHP.pdf)
- [ANO-06] <http://ns1.cic.ac.id/~ebook/ebook/adm/myebook/0062.pdf>
Diakses terakhir 15 Februari 2007.
- [ANO-07] Anonymous, 2007, "PHP", <http://en.wikipedia.org/wiki/Php> , Diakses terakhir 15 Februari 2007.
- [AGU-04] Agung, Gregorius, 2004, "WAP Programming dengan WML", Yogyakarta : Panduan.
- [IEC-08] International Engineering Consortium, 2008 "Global System for Mobile Communication (GSM)"
<http://burnsidetelecom.com/whitepapers/gsm.pdf>
- [LAS-05] Lasmono, Farid, 2005, "Aplikasi Pemesanan Tiket Bus Malam Melalui Jaringan GPRS dengan Layanan WAP", Universitas Brawijaya, Fakultas Teknik Jurusan Teknik Elektro.
- [NUG-05] Nugroho, Bunafit, 2005, "Pengembangan Program WAP dengan WML & PHP", Gava Media, Yogyakarta.
- [STI-09] Stiawan, Deris, 2009, " Teknologi GSM dan CDMA",
http://www.saranatel.com/files/gsm_cdma.pdf
- [SUJ-05] Suja, Iman, 2005, " Pemrograman SQL dan Database Server MySQL", Yogyakarta : Andi Offset.
- [SUT-05] Sutarman, 2005, "Membangun Aplikasi Web dengan PHP dan MySQL", Yogyakarta : Graha ilmu.
- [SUY-04] Suyanto, Asep Herman, 2004, " Basis Data dan DBMS"
<http://asep-hs.web.ugm.ac.id/Artikel/BASIS%20DATA%20DAN%20DBMS/BASIS%20DATA%20DAN%20DBMS.pdf>
- [SYA-04] Syahputra, Andry, 2004, "Apache Web Server", Yogyakarta : Andi Offset
- [TEL-04] Telkom, STT, 2004, "Seri Penuntun Praktis Membangun Wireless Application Protocol (WAP)", Jakarta : Elex Media Komputindo