



**RANCANG BANGUN APLIKASI PELAYANAN  
ADMINISTRASI PERAWATAN SEPEDA MOTOR DI  
PT. SURAPITA UNITRANS SURABAYA**



**TUGAS AKHIR**

**Program Studi  
S1 Sistem Informasi**

**INSTITUT BISNIS  
DAN INFORMATIKA**

**stikom  
SURABAYA**

**Oleh:  
Abner Mahendra  
09.41010.0145**

---

**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA  
2015**

**RANCANG BANGUN APLIKASI PELAYANAN ADMINISTRASI  
PERAWATAN SEPEDA MOTOR DI PT. SURAPITA UNITRANS  
SURABAYA**

**SKRIPSI**



Diajukan sebagai salah satu syarat untuk menyelesaikan  
Program Sarjana Komputer

INSTITUT BISNIS  
DAN INFORMATIKA

stikom

Oleh:

Nama : Abner Mahendra

NIM : 09.41010.0145

Program : S1 (Strata Satu)

Jurusan : Sistem Informasi

**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA**

**2015**



*Jangan patah semangat walau apapun yang terjadi,  
Jika kita menyerah maka habislah sudah*

INSTITUT BISNIS  
DAN INFORMATIKA

stikom

SURABAYA



*Kupersembahkan untuk My Father Jesus Christ, Mama, Alm. Papa, Nenek  
Beserta semua orang yang telah mendukung*

INSTITUT BISNIS  
DAN INFORMATIKA

STIKOM

SURABAYA

## DAFTAR ISI

	Halaman
ABSTRAK .....	vi
KATA PENGANTAR .....	vii
DAFTAR ISI .....	ix
DAFTAR TABEL .....	xii
DAFTAR GAMBAR .....	xx
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah .....	3
1.3 Pembatasan Masalah .....	4
1.4 Tujuan .....	4
1.5 Sistematika Penulisan.....	5
BAB II LANDASAN TEORI .....	7
2.1 Administrasi.....	7
2.2 <i>Service</i> .....	7
2.3 <i>Data Flow Diagram</i> .....	8
2.4 <i>Entity Relationship Diagram</i> .....	11
2.5 <i>System Development Life Cycle</i> .....	12
2.6 <i>Shortest Job First</i> .....	15
2.7 PT. Surapita Unitrans Surabaya.....	16
2.8 <i>SMS Gateway</i> .....	16
2.9 <i>Hyper Text Markup Language</i> .....	18
2.10 <i>Personal Home Page Tools</i> .....	18

	Halaman
2.10.1 Definisi <i>Personal Home Page Tools</i> .....	18
2.10.2 Alasan Mempelajari PHP .....	19
2.10.3 Cara Kerja Keperluan <i>Software</i> .....	20
2.11 <i>MySQL</i> .....	21
<b>BAB III ANALISIS DAN PERANCANGAN SISTEM</b> .....	<b>23</b>
3.1 Analisis Sistem .....	23
3.1.1 Identifikasi Permasalahan.....	23
3.1.2 Analisis Permasalahan. ....	24
3.2 Perancangan Sistem.....	30
3.2.1 <i>System Flow</i> Pelayanan <i>Administrasi Perawatan</i> .....	30
3.2.2 <i>Diagram HIPO</i> .....	46
3.2.3 <i>Data Flow Diagram (DFD)</i> .....	48
3.2.4 <i>Entity Relationship Diagram (ERD)</i> .....	58
3.2.5 <i>Struktur Database</i> .....	61
3.2.6 <i>Desain Input Output</i> .....	69
<b>BAB IV IMPLEMENTASI DAN EVALUASI</b> .....	<b>91</b>
4.1 Kebutuhan Sistem.....	91
4.1.1 <i>Software dan Hardware untuk Server</i> .....	91
4.1.2 <i>Software dan Hardware Client</i> .....	91
4.2 Implementasi Sistem.....	92
4.3 Evaluasi Sistem .....	112
4.3.1 Uji Coba Form .....	112
4.3.2 Hasil Evaluasi .....	138

	Halaman
BAB V PENUTUP .....	141
5.1 Kesimpulan .....	141
5.2 Saran .....	141
DAFTAR PUSTAKA .....	142

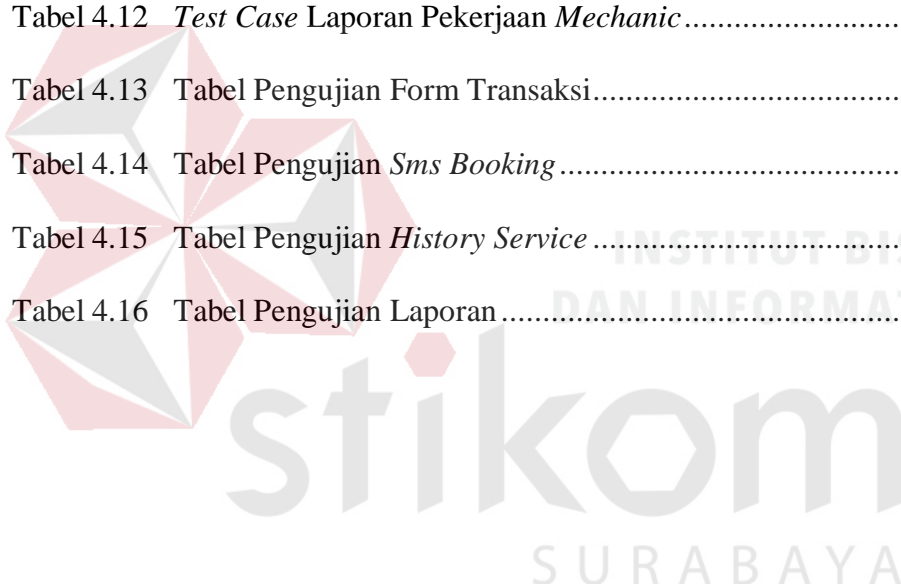


## DAFTAR TABEL

	Halaman
Tabel 3.1 Antrian .....	25
Tabel 3.2 Contoh Jadwal <i>Service</i> .....	25
Tabel 3.3 Contoh Penjadwaan <i>Shortest Job First</i> .....	25
Tabel 3.4 <i>User</i> .....	61
Tabel 3.5 Detail <i>User</i> .....	62
Tabel 3.6 Jenis Kendaraan.....	62
Tabel 3.7 Kendaraan .....	63
Tabel 3.8 Jenis <i>Service</i> .....	63
Tabel 3.9 <i>Mechanic</i> .....	64
Tabel 3.10 <i>Sparepart</i> .....	64
Tabel 3.11 Hari.....	65
Tabel 3.12 Perkiraan Waktu.....	65
Tabel 3.13 Jadwal.....	66
Tabel 3.14 Transaksi.....	66
Tabel 3.15 Transaksi Jasa.....	67
Tabel 3.16 Transaksi Part.....	68
Tabel 3.17 <i>Sms Booking</i> .....	68
Tabel 3.18 <i>Sms Reminder</i> .....	69
Tabel 4.1 <i>Test Case User Login</i> .....	113
Tabel 4.2 <i>Test Case Menu Master</i> .....	115
Tabel 4.3 <i>Test Case Penjadwalan Reguler</i> .....	118
Tabel 4.4 <i>Test Case Penjadwalan Booking</i> .....	122



	Halaman
Tabel 4.5 <i>Test Case</i> Penjadwalan <i>Reguler</i> dan <i>Booking</i> .....	127
Tabel 4.6 <i>Test Case Sms Booking</i> .....	130
Tabel 4.7 <i>Test Case History Service</i> .....	132
Tabel 4.8 <i>Test Case Laporan Service</i> .....	133
Tabel 4.9 <i>Test Case Kerusakkan Service</i> .....	134
Tabel 4.10 <i>Test Case</i> Pendapatan Jasa .....	135
Tabel 4.11 <i>Test Case</i> Pendapatan <i>Sparepart</i> .....	136
Tabel 4.12 <i>Test Case</i> Laporan Pekerjaan <i>Mechanic</i> .....	137
Tabel 4.13 Tabel Pengujian Form Transaksi .....	138
Tabel 4.14 Tabel Pengujian <i>Sms Booking</i> .....	139
Tabel 4.15 Tabel Pengujian <i>History Service</i> .....	139
Tabel 4.16 Tabel Pengujian Laporan .....	139



## DAFTAR GAMBAR

	Halaman
Gambar 2.1 Simbol <i>External Entity</i> .....	9
Gambar 2.2 Simbol <i>Data Flow</i> .....	9
Gambar 2.3 Simbol <i>Process</i> .....	10
Gambar 2.4 Simbol <i>Data Store</i> .....	10
Gambar 2.5 <i>System Development Life Chycle (SDLC) waterfall</i> .....	13
Gambar 3.1 <i>Document Flow</i> Pelayanan <i>Administrasi</i> Perawatan Sepeda Motor .....	28
Gambar 3.2 <i>System Flow</i> Pendaftaran Pelanggan (User) Baru .....	32
Gambar 3.3 <i>System Flow</i> Jenis Kendaraan .....	33
Gambar 3.4 <i>System Flow</i> Data Kendaraan.....	34
Gambar 3.5 <i>System Flow</i> Jenis <i>Service</i> .....	35
Gambar 3.6 <i>System Flow</i> Data <i>Mechanic</i> .....	36
Gambar 3.7 <i>System Flow</i> Data <i>Sparepart</i> .....	37
Gambar 3.8 <i>System Flow</i> Data hari.....	38
Gambar 3.9 <i>System Flow</i> Perkiraan Waktu.....	39
Gambar 3.10 <i>System Flow</i> Detail <i>User</i> .....	40
Gambar 3.11 <i>System Flow</i> Pendaftaran <i>Booking Service</i> .....	42
Gambar 3.12 <i>System Flow</i> Transaksi <i>Administrasi</i> .....	45
Gambar 3.13 <i>System Flow</i> Transaksi <i>Administrasi</i> .....	46
Gambar 3.14 Diagram <i>HIPO</i> .....	47
Gambar 3.15 <i>Context Diagram</i> .....	49
Gambar 3.16 <i>DFD level 0</i> Melakukan Proses <i>Maintenance Master</i> .....	51

	Halaman
Gambar 3.17 <i>DFD level 0</i> Melakukan Proses Transaksi .....	51
Gambar 3.18 <i>DFD level 0</i> Melakukan Proses Pelaporan.....	52
Gambar 3.19 <i>DFD Level 1</i> Mengelola Maintenance Master .....	53
Gambar 3.20 <i>DFD Level 1</i> Mengelola Transaksi.....	54
Gambar 3.21 <i>DFD Level 1</i> Membuat Laporan.....	56
Gambar 3.22 <i>DFD Level 2</i> Melakukan Proses Penjadwalan .....	57
Gambar 3.23 <i>Conceptual Data Model</i> .....	58
Gambar 3.24 <i>Physical Data Model</i> .....	60
Gambar 3.25 Form <i>Login</i> .....	70
Gambar 3.26 Form Menu Utama.....	71
Gambar 3.27 Form <i>User</i> (Pelanggan) .....	72
Gambar 3.28 Form Jenis Kendaraan.....	73
Gambar 3.29 Form Kendaraan .....	74
Gambar 3.30 Form Jenis <i>Service</i> .....	75
Gambar 3.31 Form <i>Mechanic</i> .....	76
Gambar 3.32 Form <i>Sparepart</i> .....	77
Gambar 3.33 Form Hari .....	78
Gambar 3.34 Form Perkiraan .....	79
Gambar 3.35 Display Detail <i>User</i> .....	80
Gambar 3.36 Form Detail <i>User</i> .....	81
Gambar 3.37 Form <i>Input</i> Transaksi Nopol .....	82
Gambar 3.38 Form <i>Input</i> Transaksi Administrasi .....	83
Gambar 3.39 Jadwal <i>Service</i> .....	84

	Halaman
Gambar 3.40 Desain <i>Work Order</i> .....	84
Gambar 3.41 Desain Nota Transaksi .....	85
Gambar 3.42 Desain Laporan <i>Service</i> .....	86
Gambar 3.43 Desain Laporan Kerusakan <i>Service</i> .....	87
Gambar 3.44 Desain Laporan Pendapatan Jasa <i>Service</i> .....	88
Gambar 3.45 Desain Laporan Pendapatan <i>Sparepart</i> .....	89
Gambar 3.46 Desain Laporan Pekerjaan <i>Mechanic</i> .....	90
Gambar 4.1 Form <i>Login</i> .....	94
Gambar 4.2 Form Menu Utama.....	95
Gambar 4.3 Form <i>Input</i> Master <i>User</i> (Pelanggan) .....	96
Gambar 4.4 Form <i>Input</i> Jenis Kendaraan.....	96
Gambar 4.5 Form <i>Input</i> Kendaraan .....	97
Gambar 4.6 Form <i>Input</i> Jenis <i>Service</i> .....	98
Gambar 4.7 Form <i>Input</i> <i>Mechanic</i> .....	99
Gambar 4.8 Form <i>Input</i> <i>Sparepart</i> .....	99
Gambar 4.9 Display Hari Operasional .....	100
Gambar 4.10 Form <i>Input</i> Perkiraan Waktu <i>Service</i> .....	101
Gambar 4.11 Display Detail <i>User</i> .....	101
Gambar 4.12 Detail <i>User</i> .....	102
Gambar 4.13 Form <i>Input</i> Detail <i>User</i> Penambahan Kendaraan .....	103
Gambar 4.14 Form <i>Input</i> Nopol .....	104
Gambar 4.15 Form <i>Input</i> Transaksi Administrasi .....	105

	Halaman
Gambar 4.16 Nota Tagihan .....	106
Gambar 4.17 Halaman Jadwal <i>Service</i> .....	107
Gambar 4.18 Laporan <i>Service</i> .....	108
Gambar 4.19 Laporan Kerusakkan <i>Service</i> .....	109
Gambar 4.20 Laporan Pendapatan Jasa <i>Service</i> .....	110
Gambar 4.21 Laporan Pendapatan <i>Sparepart</i> .....	111
Gambar 4.22 Laporan Pekerjaan <i>Mechanic</i> .....	112
Gambar 4.23 Tampilan Menu <i>Login</i> .....	114
Gambar 4.24 Menu Utama <i>Admin</i> .....	114
Gambar 4.25 Menu Utama <i>Kasir</i> .....	114
Gambar 4.26 Menu Utama Kepala Divisi .....	114
Gambar 4.27 Menu Utama Pelanggan .....	114
Gambar 4.28 <i>Message Box</i> <i>Sparepart</i> .....	116
Gambar 4.29 Peringatan pada Form <i>Sparepart</i> .....	116
Gambar 4.30 Form Pemberitahuan pada <i>Sparepart</i> .....	116
Gambar 4.31 Pernjadwalan <i>Reguler</i> .....	117
Gambar 4.32 Tampilan Transaksi <i>Reguler</i> .....	119
Gambar 4.33 Detail Transaksi <i>Reguler</i> .....	119
Gambar 4.34 Tampilan Transaksi <i>Reguler</i> berdasarkan Pendeknya Proses ..	120
Gambar 4.35 Detail Transaksi <i>Reguler</i> berdasarkan Pendeknya Proses .....	120
Gambar 4.36 Input Nopol Transaksi <i>Reguler</i> pada hari yang sama .....	120
Gambar 4.37 Peringatan melebihi jam kerja pada transaksi <i>Reguler</i> .....	121
Gambar 4.38 Penjadwalan <i>Booking</i> .....	121

	Halaman
Gambar 4.39 Tampilan Transaksi <i>Booking</i> .....	123
Gambar 4.40 Detail Transaksi <i>Booking</i> .....	123
Gambar 4.41 Tampilan <i>Booking</i> berdasarkan ketersediaan <i>Mechanic</i> .....	124
Gambar 4.42 Detail <i>Booking</i> berdasarkan ketersediaan <i>Mechanic</i> .....	124
Gambar 4.43 Peringatan <i>Input</i> Nopol pada hari yang sama.....	125
Gambar 4.44 Peringatan Slot ketersediaan <i>Mechanic</i> .....	125
Gambar 4.45 Penjadwalan <i>Reguler</i> dan <i>Booking</i> .....	126
Gambar 4.46 Tampilan Transaksi <i>Reguler</i> dan <i>Booking</i> .....	128
Gambar 4.47 Detail Transaksi <i>Reguler</i> dan <i>Booking</i> .....	128
Gambar 4.48 Tampilan transaksi prioritas pendeknya proses.....	129
Gambar 4.49 Detail transaksi prioritas pendeknya proses.....	129
Gambar 4.50 Input Nopol Transaksi pada hari yang sama .....	130
Gambar 4.51 Peringatan melebihi jam kerja .....	130
Gambar 4.52 <i>Testing Sms Gateway</i> .....	131
Gambar 4.53 Detail <i>Testing Sms Gateway</i> .....	131
Gambar 4.54 <i>Testing Sms Gateway</i> tidak adanya <i>notification</i> .....	132
Gambar 4.55 <i>Testing History Service</i> .....	133
Gambar 4.56 <i>Testing Laporan Service</i> .....	134
Gambar 4.57 <i>Testing Laporan Kerusakan Service</i> .....	135
Gambar 4.58 <i>Testing Laporan Pendapatan Jasa Service</i> .....	136
Gambar 4.59 <i>Testing Laporan Pendapatan Sparepart</i> .....	137
Gambar 4.60 <i>Testing Laporan Pekerjaan Mechanic</i> .....	138
Gambar 4.61 <i>Testing Interface</i> Transaksi .....	139

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Perkembangan teknologi yang semakin maju menyebabkan adanya persaingan antar dunia usaha yang semakin ketat. Begitu juga kesuksesan suatu perusahaan tidak hanya ditentukan dari apa yang dijual, melainkan bagaimana cara suatu perusahaan untuk menjual barang atau jasa yang dihasilkan dan ditawarkan juga sangat penting. Pada proses-proses tersebut belum menjamin bahwa pelanggan akan kembali melakukan bisnis dengan perusahaan. Salah satu cara yang dapat digunakan adalah menjaga hubungan baik antara perusahaan dengan pelanggan.

PT. Surapita Unitrans merupakan sebuah perusahaan yang salah satunya menangani proses perawatan sepeda motor merk Kawasaki. Perusahaan ini berlokasi di Jl. Jemur Sari No. 156 – 158, Surabaya. Perusahaan ini memberikan jasa perawatan sepeda motor kepada pelanggan dengan tujuan selalu berusaha memberikan nilai lebih kepada pelanggan dalam bentuk kepuasan dan pelayanan dengan baik. Tiap hari perusahaan rata-rata menangani service sebanyak 40 pelanggan dan rata-rata segmentasi pelanggan sebagian besar karyawan yang artinya kalangan menengah keatas.

Saat ini, standar operasional dan prosedur pelayanan administrasi perawatan sepeda motor yang diterapkan oleh perusahaan dimulai dari pelanggan datang mengisi data kendaraan menggunakan formulir pendaftaran serta memilih layanan perawatan sepeda motor di bagian *customer service* (CS), lalu CS

membuat lembar kerja kembali dan mengurutkan berdasarkan antrian. Lembar kerja yang sudah diurutkan oleh CS diberikan kepada *chief mechanic* (CM). CM memberikan lembar kerja tersebut kepada *mechanic* untuk dilakukan *service* (perawatan sepeda motor). Setelah perawatan selesai *mechanic* membuat laporan pengerjaan dan diberikan kepada CM. Berdasarkan laporan ini CM melakukan *control* terhadap implementasi dari laporan kinerja tersebut. Setelah selesai Kasir akan mencatat total biaya *service*, kasir akan melaporkan kembali kepada *customer service* untuk dilakukan pengecekan. Selanjutnya *customer service* akan membuat laporan untuk dilaporkan kepada *After Sale* setiap seminggu sekali.

Permasalahan yang muncul dari penerapan standar layanan administrasi pada saat ini adalah pelanggan dalam melakukan *service* tidak mendapatkan informasi penjadwalan *service* pada layanan *service* tersebut, sehingga pelanggan tidak dapat mengetahui jadwal antrian *service* yang terjadi pada saat itu. Sedangkan pelanggan yang tidak mempunyai waktu harus datang lebih awal untuk segera dilakukan *service* kendaraan, sehingga diperlukan sistem yang menangani *booking service* agar pelanggan yang mempunyai kesibukan yang padat dapat memilih hari dan jam sesuai waktu yang mereka tentukan. Permasalahan lain adalah pelanggan yang telah melakukan *service* tidak mendapatkan sebuah informasi *service* apa saja yang pernah dilakukan, dan siapa *mechanic* yang pernah menangani *service* sepeda motor pelanggan tersebut, sehingga dibutuhkan sebuah *history service* untuk pelanggan. Permasalahan yang juga terjadi di perusahaan adalah admin harus merekap kembali untuk membuat laporan kepada bagian *After Sale* dengan menggunakan *Ms Office*.



Mengatasi permasalahan tersebut dibutuhkan sebuah aplikasi yang dapat melakukan pendaftaran *service* secara otomatis untuk mengatur antrian pelanggan. Serta sistem dapat mengatur lembar kerja atau *work order* untuk *mechanic*. Sistem juga dilengkapi pemesanan antrian *service* (*Booking Service*) sehingga pelanggan juga dapat melakukan pemesanan waktu dan jam sesuai keinginan pelanggan. Sedangkan dalam hal pemberian informasi *service* kepada pelanggan akan dibuat *history service* yang akan dilihat oleh pelanggan secara langsung. Hal ini menyebabkan pelanggan dapat mengetahui *service* apa saja yang telah dilakukan. Selanjutnya, sistem juga akan dilengkapi dengan pembuatan pelaporan yaitu laporan data master, laporan *service*, laporan kerusakan *service*, laporan pendapatan jasa, laporan pendapatan *sparepart*, dan laporan pekerjaan *mechanic* yang akan diinformasikan secara rutin sesuai dengan kebutuhan admin di perusahaan tersebut. Penggunaan aplikasi ini berbasis *website* karena pelanggan juga dapat dengan mudah mengakses aplikasi ini dari rumah. Dengan adanya sistem tersebut nantinya dapat memberikan solusi untuk membantu permasalahan pelayanan administrasi perawatan sepeda motor sehingga dapat membantu meningkatkan hubungan antara pelanggan dengan perusahaan.

## **1.2 Perumusan Masalah**

Berdasarkan latar belakang, maka perumusan masalahnya adalah Bagaimana Merancang dan Membangun Aplikasi Pelayanan Administrasi Perawatan Sepeda Motor di PT. Surapita Unitrans Surabaya

### 1.3 Pembatasan Masalah

Berdasarkan rumusan masalah yang telah disebutkan diatas, adapun batasan-batasan masalah dalam Merancang dan Membangun Aplikasi Pelayanan Administrasi Perawatan Sepeda Motor di PT. Surapita Unitrans Surabaya yaitu :

1. Aplikasi ini dalam bentuk website dan hanya membahas layanan administrasi perawatan sepeda motor.
2. Aplikasi ini tidak membahas pembelian *sparepart* ke pemasok.
3. Aplikasi ini dilengkapi pemesanan antrian service
4. Penggunaan aplikasi *SMS Gateway* hanya membahas konfirmasi kedatangan pelanggan
5. Waktu tunggu keterlambatan pelanggan setelah 15 menit waktu *booking* dan jika melebihi 15 menit transaksi akan dihapus.
6. Tidak menangani kegagalan pengiriman pesan diasumsikan pesan *sms* selalu terkirim.
7. Bahasa pemrograman yang digunakan adalah *PHP* (*Hypertext Preprocessor*), dan *MySQL Server* sebagai basis datanya.

### 1.4 Tujuan

Dengan melihat perumusan masalah di atas, maka tujuan yang hendak dicapai adalah menghasilkan Rancang Bangun Aplikasi Pelayanan Administrasi Perawatan Sepeda Motor di PT. Surapita Unitrans Surabaya.

## 1.5 Sistematika Penulisan

Di dalam penyusunan laporan tugas akhir ini secara sistematis diatur dan disusun dalam lima bab, yang masing-masing terdiri dari beberapa sub bab. Adapun urutan dari bab pertama sampai bab terakhir adalah sebagai berikut:

### **BAB I            PENDAHULUAN**

Bab ini membahas tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan pembuatan sistem, manfaat bagi penggunaanya, serta sistematika penulisan laporan.

### **BAB II           LANDASAN TEORI**

Bab ini membahas mengenai berbagai macam teori yang mendukung dalam pembuatan Rancang Bangun Aplikasi Pelayanan Administrasi Perawatan Sepeda Motor di PT. Surapita Unitrans Surabaya. Teori pendukung tersebut antara lain : penjelasan tentang *administrasi, service, data flow diagram, entity relationship diagram, system development life cycle, sms gateway, hyper text makrup language, personal home page tools, mySQL*, dan metode *shortest job first* yang digunakan sistem untuk menghasilkan sebuah penjadwalan perawatan sepeda motor .

### **BAB III          ANALISIS DAN PERANCANGAN SISTEM**

Bab ini membahas analisis dan perancangan sistem. Analisis berisi penjelasan dari timbulnya masalah atau identifikasi permasalahan beserta penyelesaiannya atau analisis permasalahan , sedangkan perancangan sistem berisi *Document Flow, System Flow, HIPO*,

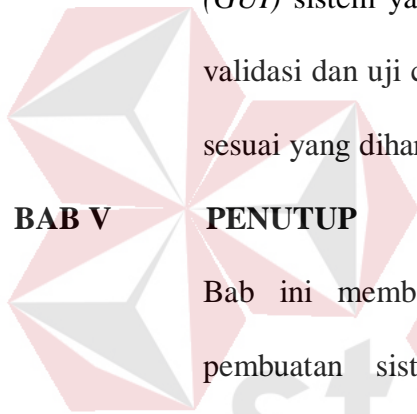
*Data Flow Diagram, Entity Relationship Diagram, Struktur Database, dan Desain Input / Output.*

#### **BAB IV IMPLEMENTASI DAN EVALUASI SISTEM**

Bab ini membahas tentang kebutuhan perangkat lunak, perangkat keras, implementasi dan evaluasi sistem. Implementasi ini mengacu pada perancangan desain sistem yang telah dibuat dan berfokus memberikan keputusan kelayakan pinjaman nasabah. Dalam implementasi ini juga berisi penjelasan *Graphical User Interface (GUI)* sistem yang telah dibuat. Sedangkan evaluasi sistem berisi validasi dan uji coba sistem agar terhindar dari *error* serta berjalan sesuai yang diharapkan.

#### **BAB V PENUTUP**

Bab ini membahas tentang kesimpulan yang diperoleh dari pembuatan sistem ini serta saran yang bertujuan untuk pengembangan sistem dimasa yang akan datang.



## BAB II

### LANDASAN TEORI

#### 2.1 Administrasi

Secara umum pengertian *administrasi* dikenal sebagai suatu kegiatan yang meliputi proses pencatatan, pengiriman surat, pembukuan ringan serta berbagai macam kegiatan yang dimaksudkan untuk menyediakan informasi agar dapat memperoleh dengan mudah informasi jika dibutuhkan kembali. Asal usul kata atau etimologi dari administrasi sendiri bermula dari bahasa latin yaitu “ad” serta “*ministrare*” yang artinya pemberian jasa atau bantuan, yang dalam bahasa inggris disebut “*Administration*” artinya “*To Serve*”, yaitu melayani dengan cara yang paling baik (Nuraida, 2008).

#### 2.2 Service

*Service* adalah merupakan suatu pelayanan yang diberikan kepada pelanggan untuk digunakan sebagai tujuan meningkatkan kepuasan dan kenyamanan pelanggan (Barata, 2008)

Satu-satunya jalan untuk mempertahankan agar organisasi atau perusahaan selalu didekati dan diingat pelanggan adalah dengan cara mengembangkan pola layanan terbaik, antara lain dengan cara sebagai berikut (Barata, 2008):

1. Berupaya menyediakan kebutuhan perusahaan sesuai dengan keinginan atau lebih dari yang diharapkannya, dan ;

2. Memperhatikan perkembangan kebutuhan dan keinginan para pelanggan dari waktu ke waktu, untuk kemudahan mengantisipasinya.

Jadi dimasa saat ini, disamping menonjolkan kualitas produk (barang dan jasa) sebagai wujud layanan, para pelaku bisnis menonjolkan juga pola layanan lainnya sebagai penunjang untuk memberikan perhatian serta keakraban kepada pelanggannya, yang kemudian dikenal sebagai kepedulian terhadap pelanggan (*after sale*).Sebaiknya pelayanan dilakukan untuk mewujudkan manfaat terbaik bagi semua pihak, yaitu dengan mengembangkan strategi pelayanan.

Pelayanan dibagi menjadi dua macam pelayanan antara lain sebagai berikut (Yoeti, 1999) :

1. Layanan Internal

Orang-orang yang terlibat dalam proses produksi dan jasa yang kita hasilkan. Maka pelayanan internal pelanggan tidak lain adalah karyawan, *Admin Service, Mechanic, Chief Mechanic, Kasir, Kepala Bagian After Sale* (Yoeti, 1999).

2. Layanan Eksternal

Layanan Eksternal secara relative mudah diidentifikasi karena berada diluar organisasi suatu perusahaan, yang menerima barang atau jasa dari perusahaan (Yoeti, 1999).

### 2.3 *Data Flow Diagram*

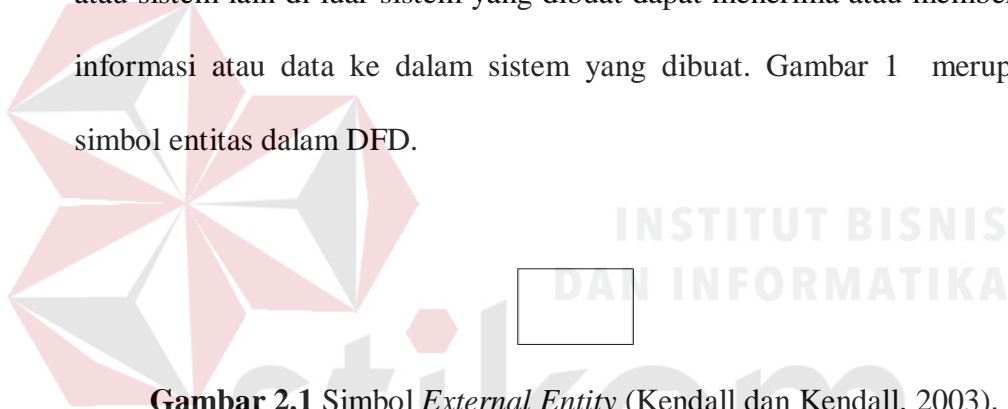
*Data Flow Diagram* (DFD) sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik di tempat data tersebut mengalir. DFD

merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur dan dapat mengembangkan arus data di dalam sistem dengan terstruktur dan jelas.

DFD fokus pada aliran data dari dan ke dalam sistem serta memproses data yang mengalir tersebut (Kendall dan Kendall, 2003). Simbol-simbol dasar dalam DFD yaitu:

a. *External Entity*

Suatu *ExternalEntity* atau entitas merupakan orang, kelompok, departemen, atau sistem lain di luar sistem yang dibuat dapat menerima atau memberikan informasi atau data ke dalam sistem yang dibuat. Gambar 1 merupakan simbol entitas dalam DFD.



**Gambar 2.1** Simbol *External Entity* (Kendall dan Kendall, 2003).

b. *Data Flow*

*Data Flow* atau aliran data disimbolkan dengan tanda panah. *DataFlow* menunjukkan arus data atau aliran data yang menghubungkan dua proses atau entitas dengan proses. Gambar 2 merupakan simbol *DataFlow*.

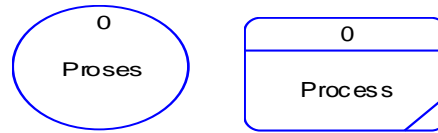


**Gambar 2.2** Simbol *Data Flow* (Kendall dan Kendall, 2003).

c. *Process*

Suatu *Process* meliputi beberapa tindakan atau sekelompok tindakan dari arus data yang masuk untuk dijalankan atau diproses agar menghasilkan arus

data yang akan keluar dari proses. Gambar 3 merupakan simbol *Process* dalam model Demarco & Yurdon dan Gane & Sarson.

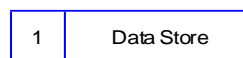


**Gambar 2.3** Simbol *Process* (Kendal dan Kendall, 2003)

d. *Data Store*

*DataStore* adalah simbol yang digunakan untuk melambangkan proses penyimpanan data. Suatu nama perlu diberikan pada *DataStore* untuk menunjukkan nama dari *file*-nya. Gambar 4 merupakan simbol *file* penyimpanan/*Data Store* yang dapat berupa hal-hal sebagai berikut, sebagai gambaran:

- 1) Suatu file atau *Database* di sistem komputer.
- 2) Suatu arsip atau catatan manual.
- 3) Suatu tabel acuan manual.



**Gambar 2.4** Simbol *Data Store* (Kendal dan Kendall, 2003)

Berikut ini adalah urutan langkah bagaimana menggambarkan suatu sistem pada DFD:

1. *Context Diagram*

*Context Diagram* merupakan langkah pertama dalam pembuatan *Data Flow Diagram*. Pada *Context Diagram* dijelaskan sistem apa yang dibuat dan



*Entity* apa saja yang digunakan. Dalam *Context Diagram* harus ada arus data yang masuk dan arus data yang keluar.

## 2. *Data Flow Diagram Level 0*

DFD Level 0 adalah langkah selanjutnya setelah *Context Diagram*. Hal yang digambarkan dalam Diagram Level 0 ini adalah proses utama dari sistem serta hubungan *Entity*, *Process*, *Data Flow* dan *Data Store*.

## 3. *Data Flow Diagram Level 1*

DFD Level 1 merupakan penjelasan dari DFD Level 0. Pada proses ini dijelaskan proses apa saja yang dilakukan pada setiap proses yang terdapat di DFD Level 0

## 2.4 *Entity Relationship Diagram*

*Entity relationship diagram* (ERD) adalah gambaran pada sistem dimana di dalamnya terdapat hubungan antara *entity* beserta relasinya. *Entity* merupakan sesuatu yang ada terdefiniskan di dalam suatu organisasi.

*Entity Relationship Diagram* ini diperlukan agar dapat menggambarkan hubungan antar *entity* dengan jelas, dapat menggambarkan batasan jumlah *entity* dan partisipasi antar *entity*, mudah dimengerti pemakai dan mudah disajikan oleh perancang *database*. Untuk itu, *entity relationship diagram* dibagi menjadi dua jenis model, yaitu:

### 1) *Conceptual Data Model*

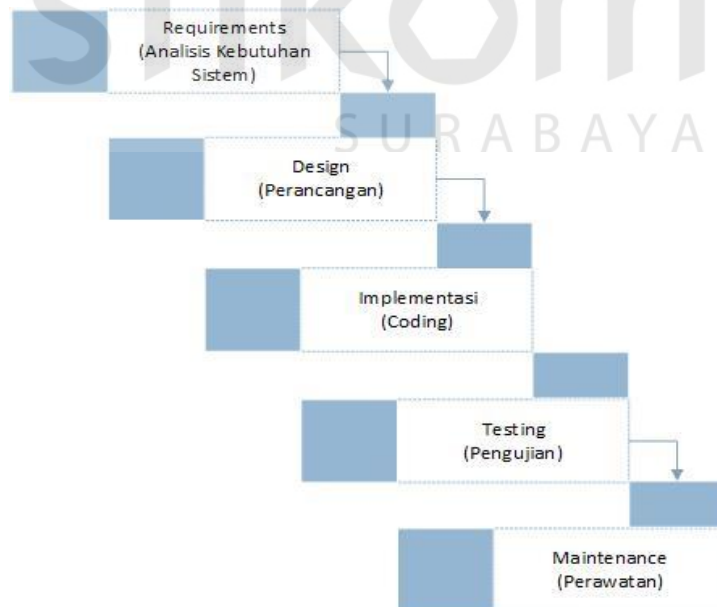
*Conceptual Data model* (CDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara konseptual.

## 2) *Physical Data Model*

*Physical Data Model* (PDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara fisik.

## 2.5 *System Development Life Cycle*

Menurut Pressman (2001), Model *System Development Life Cycle* (SDLC) ini biasa disebut juga dengan model *waterfall* atau disebut juga *classic life cycle*. Adapun pengertian dari SDLC ini adalah suatu pendekatan yang sistematis dan berurutan. Tahapan-tahapannya adalah *Requirements* (analisis sistem), *Analysis* (analisis kebutuhan sistem), *Design* (perancangan), *Coding* (implementasi), *Testing* (pengujian) dan *Maintenance* (perawatan). Model ini memungkinkan proses pengembangan lebih terlihat. Hal ini dikarenakan bentuknya yang bertingkat ke bawah dari satu fase ke fase lainnya, model ini dikenal dengan model *waterfall*, seperti terlihat pada gambar 5 berikut ini:



**Gambar 2.5** *System Development Life Cycle (SDLC) waterfall* (Pressman, 2001)

Penjelasan-penjelasan *System Development Life Cycle* (SDLC) adalah sebagai berikut:

a. *Requirement* (Analisis Kebutuhan Sistem)

Pada tahap awal ini dilakukan analisa guna menggali secara mendalam kebutuhan yang akan dibutuhkan. Kebutuhan ada bermacam-macam seperti halnya kebutuhan informasi bisnis, kebutuhan data dan kebutuhan user itu sendiri. Kebutuhan itu sendiri sebenarnya dibedakan menjadi tiga jenis kebutuhan. Pertama tentang kebutuhan teknologi. Dari hal ini dilakukan analisa mengenai kebutuhan teknologi yang diperlukan dalam pengembangan suatu sistem, seperti halnya data penyimpanan informasi / *database*. Kedua kebutuhan informasi, contohnya seperti informasi mengenai visi dan misi perusahaan, sejarah perusahaan, latar belakang perusahaan. Ketiga, Kebutuhan *user*. Dalam hal ini dilakukan analisa terkait kebutuhan user dan kategori user (Pressman, 2001).

b. *Design* (Perancangan)

Selanjutnya, hasil analisa kebutuhan sistem tersebut akan dibuat sebuah *design database*, DFD, ERD, antarmuka pengguna/*Graphical User Interface* (GUI) dan jaringan yang dibutuhkan untuk sistem. Selain itu juga perlu dirancang struktur datanya, arsitektur perangkat lunak, detil prosedur dan karakteristik tampilan yang akan disajikan. Proses ini menterjemahkan kebutuhan sistem ke dalam sebuah model perangkat lunak yang dapat diperkirakan kualitasnya sebelum memulai tahap implementasi (Pressman, 2001)

c. *Implementation* (Coding)

Rancangan yang telah dibuat dalam tahap sebelumnya akan diterjemahkan ke dalam suatu bentuk atau bahasa yang dapat dibaca dan diterjemahkan oleh komputer untuk diolah. Tahap ini juga dapat disebut dengan tahap implementasi, yaitu tahap yang mengkonversi hasil perancangan sebelumnya ke dalam sebuah bahasa pemrograman yang dimengerti oleh komputer. Kemudian komputer akan menjalankan fungsi-fungsi yang telah didefinisikan sehingga mampu memberikan layanan-layanan kepada penggunanya (Pressman, 2001).

d. *Testing* (Pengujian)

*Testing* dapat digunakan untuk memastikan kevalidan dalam proses *input*, sehingga dapat menghasilkan *output* yang sesuai. Pada tahap ini terdapat 2 metode pengujian perangkat yang dapat digunakan, yaitu: metode *black-box* dan *white-box*. Pengujian dengan metode *black-box* merupakan pengujian yang menekankan pada fungsionalitas dari sebuah perangkat lunak tanpa harus mengetahui bagaimana struktur di dalam perangkat lunak tersebut. Sebuah perangkat lunak yang diuji menggunakan metode *black-box* dikatakan berhasil jika fungsi-fungsi yang ada telah memenuhi spesifikasi kebutuhan yang telah dibuat sebelumnya. Pengujian dengan menggunakan metode *white-box* yaitu menguji struktur internal perangkat lunak dengan melakukan pengujian pada algoritma yang digunakan oleh perangkat lunak (Pressman, 2001).

e. *Maintenance* (Perawatan)

Tahap terakhir dari metode SDLC ini adalah *maintenance*. Pada tahap ini, jika sistem sudah sesuai dengan tujuan yang ditentukan dan dapat menyelesaikan masalah pada *service*, maka akan diberikan kepada pengguna. Setelah digunakan dalam periode tertentu, pasti terdapat penyesuaian atau perubahan sesuai dengan keadaan yang diinginkan, sehingga membutuhkan perubahan terhadap sistem tersebut. Tahap ini dapat pula diartikan sebagai tahap penggunaan perangkat lunak yang disertai dengan perawatan dan perbaikan. Perawatan dan perbaikan suatu perangkat lunak diperlukan, termasuk didalamnya adalah pengembangan, karena dalam prakteknya ketika perangkat lunak digunakan terkadang masih terdapat kekurangan ataupun penambahan fitur-fitur baru yang dirasa perlu (Pressman, 2001).

## 2.6 *Shortest Job First*

*Shortest Job First* atau yang dikenal sebagai SJF, SJF adalah suatu metode yang mengutamakan pelayanan terhadap pelanggan berdasarkan prioritas pendeknya suatu proses (Hariningsih, 2003). Jika panjangnya sebuah proses tidak dapat dijadwalkan akan digantikan dengan proses yang terpendek, sedangkan proses yang panjang akan dijadwalkan kembali hingga masuk dalam proses penjadwalan. Jika penjadwalan pada hari tersebut proses terpanjang masih tetap tidak bisa dijadwalkan maka order tersebut akan dikembalikan kepada pelanggan. Didalam SJF terdapat hitungan waktu tunggu kepada pelanggan dimana mempunyai beberapa istilah antara lain:

- a. LE = Lama Eksekusi
- b. SE = Selesai Eksekusi
- c. WT = Waktu Tunggu
- d. Diketahui WT = SE
- e. n = banyaknya pelanggan dalam sistem antrian

Terdapat beberapa penghitungan waktu tunggu dengan menggunakan metode Shortest Job First antara lain :

- 1) Penghitungan waktu tunggu sebelum menerima pelayanan

$$WT = LE + SE \dots\dots\dots (1)$$

- 2) Rata-rata waktu tunggu sebelum menerima pelayanan

$$WT \text{ rata-rata} = LE_1 + LE_2 + LE_3 + LE_4 + LE_5 / n \dots\dots\dots (2)$$

## 2.7 PT. Surapita Unitrans Surabaya

PT. Surapita Unitrans merupakan sebuah perusahaan yang salah satunya menangani proses *service* sepeda motor merk kawasaki. Perusahaan ini berlokasi di Jl. Jemur Sari No. 156 – 158, Surabaya. Perusahaan ini memberikan jasa *service* kepada pelanggan dengan tujuan selalu berusaha memberikan nilai lebih kepada pelanggan dalam bentuk kepuasan dan pelayanan dengan baik.

## 2.8 SMS Gateway

*SMS Gateway* adalah pintu gerbang bagi penyebaran informasi dengan menggunakan *SMS* (Tarigan, 2013). Fasilitas ini dipakai untuk mengirim dan menerima pesan dalam bentuk teks ke dan dari sebuah ponsel. (Indira dan Prianka, 2004). Menurut Soerowirdjo, beberapa karakteristik SMS adalah :

1. Sebuah pesan singkat terdiri atas 160 karakter yang terdiri dari atas huruf atau angka. Juga dapat mendukung pesan *non-text*, seperti format binary.
2. Prinsip kerjanya adalah “menyimpan” dan “menyampaikan” pesan (*store and forward message*). Dengan kata lain pesan tidak langsung dikirimkan kepenerima, tetapi disimpan dahulu di SMS Centre.
3. Memiliki ciri dalam hal konfirmasi pengiriman pesan, yaitu pesan yang dikirimkan tidak secara sederhana dikirimkan dan dipercayai akan disampaikan dengan selamat. Namun pengirim pesan dapat pula menerima pesan balik yang memberitahukan apakah pesan telah terkirim atau gagal.
4. Pesan dapat dikirim dan diterima secara simultan dengan panggilan jenis layanan GSM lain.

### **2.8.1 Perangkat yang dibutuhkan untuk membangun sms gateway**

Membangun aplikasi *SMS Gateway* diperlukan beberapa perangkat pendukung, diantaranya (Tarigan, 2013) :

1. PC Server (Minimal : Pentium 3, 256 MB RAM, Hardisk 40 GB).
2. Modem GSM USB lengkap dengan SIM Card.
3. Driver modem GSM.
4. *SMS now*.
5. Apache + MySQL ( Paket XAMPP ).

## 2.9 *Hyper Text Markup Language*

*Hyper Text Markup Language* merupakan singkatan dari HTML. Dokumen HTML adalah merupakan file text murni yang dapat dibuat dengan editor text sembarang (Kustiyahningsih, 2011).

### 2.9.1 Struktur Dasar HTML

Elemen yang dibutuhkan untuk membuat suatu dokumen HTML dinyatakan dengan tag – tag sebagai berikut (Kustiyahningsih, 2011) :

#### 1. HTML

Setiap dokumen HTML selalu diawali dan ditutup dengan tag HTML.

#### 2. HEAD

Bagian head biasanya berisikan tag TITLE, meta tag dan semua *script java* atau bagian lain yang akan dieksekusi di *browser*. Pada bagian inilah kita memberikan *bookmark* untuk keperluan pencarian (*searching*) dengan keyword.

#### 3. BODY

Bagian body yang digunakan untuk menampilkan text, image link dan semua yang akan ditampilkan pada *web page*.

## 2.10 *Personal Home Page Tools*

### 2.10.1 Definisi *Personal Home Page Tools*

*Personal Home Page Tools* atau yang disingkat PHP, skrip ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML sehingga suatu



halaman web tidak lagi bersifat statis, namun menjadi bersifat dinamis (Kustiyaningsih, 2011).

Nugroho (2013) mengatakan bahwa PHP adalah bahasa program yang digunakan untuk membuat aplikasi berbasis web (website, blog atau aplikasi web).

Sakur (2010) mengatakan *Hypertext PreProcessor* atau dikenal dengan nama PHP merupakan salah satu aplikasi server yang sangat banyak digunakan sampai sekarang karena kemudahan dan keandalan yang dimilikinya.

Beberapa definisi diatas maka dapat diambil 4 (empat) point utama tentang PHP yaitu :

1. PHP adalah singkatan dari *Hypertext PreProcessor*.
2. PHP adalah bahasa *scripting server-side* yang artinya dijalankan pada *server* kemudian outputnya dikirim ke *client*.
3. PHP digunakan untuk membuat aplikasi web.
4. PHP mendukung banyak database (*MqSQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, dll*).

### 2.10.2 Alasan Mempelajari PHP

Beberapa alasan untuk mempelajari PHP adalah (Anhar, 2010:3) :

1. Kesederhanaan. PHP memiliki banyak sekali fungsi *build-in* untuk menangani kebutuhan standar pembuatan aplikasi Web. Tentu saja dengan adanya fungsi – fungsi tersebut proses belajar PHP terutama dalam pengembangan aplikasi akan jauh lebih mudah karena semua telah tersedia.

2. PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
3. PHP adalah bahasa *open source* yang dapat digunakan diberbagai sistem operasi. *Open source* artinya kode – kode PHP terbuka untuk umum sehingga tidak diharuskan untuk membayar biaya pembelian atas keaslian *license* yang biasanya cukup mahal.
4. Web server yang mendukung PHP dapat ditemukan dimana – mana mulai dari Apache, IIS, Lighttpd, hingga Citami dengan konfigurasi yang relative murah.
5. PHP juga dilengkapi dengan berbagai macam pendukung lain seperti support langsung ke berbagai macam database yang populer, misal : Oracle, Postgre SQL, MySQL dan lain – lain.

### 2.10.3 Cara Kerja dan Keperluan *Software*

Cara kerja PHP yaitu : pertama *client web browser* atau pengguna memakai komputer kemudian pengguna tersebut menjalankan file PHP tersebut di *web browser*, kemudian file PHP itu dikirim ke *web server*, *web server* mengirimkan lagi ke *engine PHP* atau mesin PHP (Nugroho, 2013).

*Web server* adalah aplikasi yang berfungsi untuk melayani permintaan pemanggilan alamat dari pengguna melalui *web browser*, dimana *web server* mengirimkan kembali informasi yang diminta tersebut melalui HTTP (*Hypertext Transfer Protocol*) untuk ditampilkan ke layar monitor komputer. Agar dapat mengubah *website* yang dibuat maka dibutuhkan program PHP. *Script – script* PHP tersebut yang berfungsi membuat halaman *website* menjadi dinamis.

Dinamis artinya pengunjung *web* dapat memberikan komentar, saran atau masukan pada *website* sehingga *website* yang dibuat menjadi lebih hidup karena ada komunikasi antara pengunjung dan pemberi jasa (*web master*)

## 2.11 MySQL

*MySQL* sebuah aplikasi SQL *database server* yang *multi user, multithread*. Tujuan dari *MySQL* ialah kecepatan dan mudah penggunaannya. Ini yang menyebabkan banyak aplikasi web yang berbasis *database* selalu menggunakan *MySQL* sebagai *database engine*nya.

*MySQL* adalah *database server* relasional yang gratis di bawah lisensi *General Public Lisensi* (GNU). Dengan sifatnya yang *open source*, memungkinkan juga user untuk melakukan modifikasi pada *source codenya* untuk memenuhi kebutuhan spesifik mereka sendiri. *MySQL* merupakan *database server multi user* dan *multi threaded* yang tangguh. Dengan memiliki banyak *feature MySQL* bisa bersaing dengan *database* komersial sekalipun. Tidak mengejutkan *MySQL* menjadi *database* pilihan untuk banyak pengguna PHP.

*MySQL* termasuk jenis *Relational Database Management System* (RDBMS) itulah sebabnya istilah seperti tabel, baris dan kolom digunakan pada *MySQL*. Pada *MySQL* sebuah *database* mengandung satu atau sejumlah tabel. Tabel terdiri dari sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

Menurut Marlinda (2004), *Database Management System* (DBMS) merupakan kumpulan file yang saling berkaitan dan program untuk pengelolanya. Basis Data adalah kumpulan datanya, sedang program pengelolanya berdiri

sendiri dalam suatu paket program yang komersial untuk membaca data, menghapus data, dan melaporkan data dalam basis data.

Penyusunan satu *database* digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu redundansi dan inkonsistensi data, kesulitan pengaksesan data, isolasi data untuk standarisasi, *multiple user* (banyak pemakai), *security* (masalah keamanan), masalah integrasi (kesatuan), dan masalah data *independence* (kebebasan data ).

Beberapa karakteristik yang dimiliki oleh *MySQL* server adalah:

1. *Application Program Interface* (API) untuk bahasa *C+*, *C++*, *PHP*, *Python*, *java*, dan *TCL*
2. Berkerja pada banyak *platform* sistem operasi
3. Mendukung penuh operator dan fungsi dalam sintack *SELECT* dan *WHERE* sebagai bagian *query*
4. Pembatasan hak dan *password* yang sangat fleksible dan aman serta mendukung vertifikasi dalam berdasarkan host
5. Mampu mengolah *database* yang besar.

## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

#### 3.1 Analisis Sistem

Menurut Jogiyanto (1999) analisis sistem merupakan bagian dari tahapan SDLC (*System Development Life Cycle*). Untuk dapat menganalisis sistem perlu pemahaman terhadap proses bisnis dan permasalahan yang terjadi. Maka dari itu perlu dilakukan langkah-langkah identifikasi masalah dan analisis kebutuhan pada PT. Surapita Unitrans.

##### 3.1.1 Identifikasi Permasalahan

Permasalahan yang muncul dari penerapan standar layanan administrasi pada saat ini adalah pelanggan dalam melakukan *service* tidak mendapatkan informasi penjadwalan *service* pada layanan *service* tersebut, sehingga pelanggan tidak dapat mengetahui jadwal antrian *service* yang terjadi pada saat itu. Sedangkan pelanggan yang tidak mempunyai waktu harus datang lebih awal untuk segera dilakukan *service* kendaraan, sehingga diperlukan sistem yang menangani *booking service* agar pelanggan yang mempunyai kesibukan yang padat dapat memilih hari dan jam sesuai waktu yang mereka tentukan. Permasalahan lain adalah pelanggan yang telah melakukan *service* tidak mendapatkan sebuah informasi *service* apa saja yang pernah dilakukan, dan siapa *mechanic* yang pernah menangani *service* sepeda motor pelanggan tersebut, sehingga dibutuhkan sebuah *history service* untuk pelanggan. Permasalahan yang

juga terjadi di perusahaan adalah admin harus merekap kembali untuk membuat laporan kepada bagian *After Sale* dengan menggunakan *Ms Office*.

### 3.1.2 Analisis Permasalahan

Dari permasalahan yang ada tersebut, maka berikut adalah analisis yang dibuat penulis untuk menyelesaikan permasalahan tersebut

#### 1. Analisis sistem dalam mengatur antrian penjadwalan *service*.

Sistem yang dibuat dapat melakukan pendaftaran *service* secara otomatis untuk mengatur antrian pelanggan. Antrian pelanggan menghasilkan *work order* untuk pengerjaan *mechanic*. Sistem juga dilengkapi pemesanan antrian *service* (*Booking Service*) sehingga pelanggan juga dapat melakukan pemesanan waktu dan jam sesuai keinginan pelanggan.

#### 2. Analisis sistem dalam penggunaan metode *shortest job first*.

Sistem yang dibuat ini menggunakan metode *shortest job first*, dimana metode ini mengutamakan pelayanan terhadap pelanggan berdasarkan prioritas pendeknya suatu proses. Jika panjangnya sebuah proses tidak dapat dijadwalkan akan digantikan dengan proses yang terpendek, sedangkan proses yang panjang akan dijadwalkan kembali hingga masuk dalam proses penjadwalan. Jika penjadwalan pada hari tersebut proses terpanjang tersebut masih tetap tidak bisa dijadwalkan maka order tersebut akan dikembalikan kepada pelanggan.

Misalkan terdapat 5 antrian yaitu A1, A2, A3, A4, A5 dan satu antrian berstatus *booking*. Masing-masing waktu kedatangan 5 menit, dan lama proses dari masing-masing antrian 2, 1, 2, 1, 1.

**Tabel 3.1** Antrian

Antrian	Waktu Pengerjaan
A1	2
A2	1
A3	2
A4	1
A5	1

Jika jadwal *service* seperti tabel 3.2 :

**Tabel 3.2** Contoh Jadwal *Service*

Mek1	Mek2	Mek3	Mek4	Mek5
Booking	Booking	Booking		

Maka proses penjadwalannya adalah sebagai berikut :

**Tabel 3.3** Contoh Penjadwalan *Shortest Job First*

Mek1	Mek2	Mek3	Mek4	Mek5
A2	A4	A5	A1	A3
Booking	Booking	Booking		

Proses Antrian : A2, A4, A5, A1, A3

1 , 1 , 1 , 2 , 2

- Analisis sistem dalam pemberian informasi kepada pelanggan.

Sistem yang dibuat akan menghasilkan sebuah informasi kepada pelanggan yang selama ini belum pernah diinformasikan, seperti halnya informasi *reminder* yang mengingatkan pelanggan untuk melakukan *service* secara berkala kepada pelanggan, informasi jadwal *service* agar pelanggan mengetahui

jadwal *service* secara langsung pada hari itu juga, informasi *sms approval* sebagai pengingat untuk pelanggan yang melakukan *booking service* berupa *sms* yang akan disampaikan satu jam sebelum jadwal *service* yang telah di *booking* tiba, informasi *estimasi waktu* dimana pelanggan mengetahui seberapa lama waktu mereka menunggu dan perbaiki pada saat melakukan *service*, dan juga informasi *progress service* yang akan disampaikan khusus kepada pelanggan yang meninggalkan motornya saat melakukan *service*, yang akan menginformasikan bahwa *service* motor pelanggan telah selesai dikerjakan oleh *mechanic*.

4. Analisis sistem pembuatan laporan.

Perusahaan dalam melakukan pelaporan masih menggunakan *MS Office* selama ini dan hanya sebatas transaksi *service* pada hari itu juga. Sehingga untuk mengatasi permasalahan tersebut dibutuhkan adanya sebuah sistem pelaporan yang dibuat secara terkomputerisasi, dan dapat memenuhi kebutuhan informasi-informasi dalam perusahaan tersebut antara lain antara lain : laporan data master, laporan *service*, laporan kerusakan *service*, laporan pendapatan jasa, laporan pendapatan *sparepart*, dan laporan pekerjaan *mechanic*.

5. *Document flow diagram* pelayanan administrasi perawatan sepeda motor

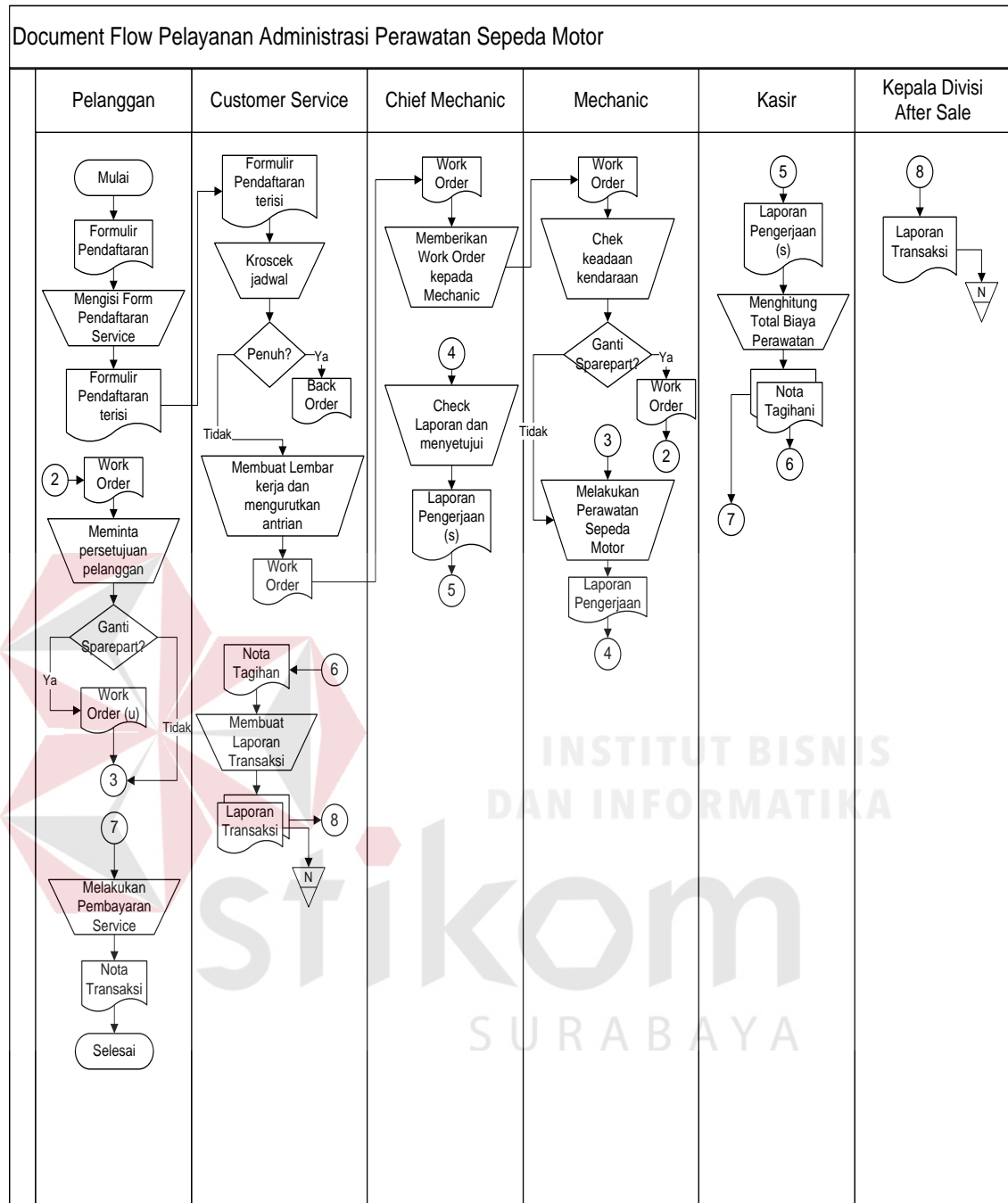
*Document Flow* merupakan bagan yang menunjukkan aliran atau arus dokumen dari satu bagian ke bagian yang lain di dalam sistem secara logika. *Document flow* juga menggambarkan tiap-tiap bagian organisasi yang terlibat dalam pengolahan dokumen di dalam tiap-tiap proses. Namun, proses yang digambarkan dalam *document flow* adalah proses manual atau proses yang



selama ini dikerjakan organisasi tanpa adanya sebuah sistem yang membantu menangani proses tersebut.

Sehubungan dengan itu dibawah ini akan digambarkan aliran dokumen pelayanan administrasi perawatan sepeda motor di PT. Surapita Unitrans Surabaya. Secara umum ada enam bagian atau entitas dalam aliran dokumen ini, yaitu pelanggan, *customer service*, *chief mechanic*, *mechanic*, *kasir*, dan kepala divisi *after sale*.





**Gambar 3.1** Document Flow Pelayanan Administrasi Perawatan Sepeda Motor

Penjelasan aliran dokumen diatas adalah sebagai berikut : Pelanggan mendapatkan formulir pendaftaran dan mengisi formulir pendaftaran untuk diberikan kepada *customer service*, setelah *customer service* menerima formulir yang telah terisi maka bagian *customer service* akan melakukan kroscek jadwal apakah jadwal yang terjadi pada hari tersebut penuh atau tidak.

Jika jadwal penuh maka order kembali kepada pelanggan. Sedangkan jika jadwal tidak penuh maka customer service akan membuat *work order* (lembar kerja) dan mengurutkan antrian sesuai kedatangan pelanggan. *Work order* akan diserahkan kepada *chief mechanic*, dan *chief mechanic* memberikan *work order* kepada *mechanic*. *mechanic* yang telah menerima *work order* akan melakukan *check* kendaraan, jika ada pergantian *sparepart* maka *mechanic* akan meminta persetujuan kepada pelanggan terlebih dahulu apakah diganti atau tidak. Sedangkan jika tidak ada pergantian *sparepart* maka *mechanic* akan melakukan perawatan sepeda motor (*service*). Setelah perawatan (*service*) selesai, *mechanic* akan membuat laporan pengerjaan apa saja yang telah dikerjakan. Laporan pengerjaan tersebut akan diberikan kepada *chief mechanic*, lalu *chief mechanic* melakukan *check* laporan pengerjaan. Setelah selesai *chief mechanic* akan menyerahkan laporan tersebut kepada kasir, dan kasir akan menghitung total biaya perawatan (*service*) yang telah dilakukan untuk menjadi nota tagihan dan di *copy* menjadi dua lembar. Nota tagihan pertama akan diberikan kepada pelanggan untuk dilakukan pembayaran. Sedangkan nota tagihan yang kedua akan diberikan kepada *customer service* untuk dibuatkan 2 lembar laporan periode. Laporan pertama untuk bagian *customer service* sendiri, sedangkan laporan kedua akan diberikan kepada kepala divisi *after sale*.

### 3.2 Perancangan Sistem

Setelah dilakukan analisis terhadap sistem, maka langkah selanjutnya adalah perancangan sistem. Perancangan sistem ini bertujuan untuk mendefinisikan kebutuhan-kebutuhan fungsional, menggambarkan aliran data dan alur sistem, dan sebagai tahap persiapan sebelum implementasi sistem. perancangan sistem ini adalah :

- a. *System Flow*.
- b. Diagram *HIPO (Hierarchy Input Process Output)*.
- c. *Data Flow Diagram (DFD)*, yang didalamnya terdapat : *context diagram, DFD Level 0, DFD Level 1 dan DFD Level 2*.
- d. *Entity Relationship Diagram (ERD)*, yang didalamnya meliputi : *Conceptual Data Model (CDM)*, dan *Physical Data Model (PDM)*.
- e. *Data Dictionary*.
- f. *Desain Input Output*.

Sistem layanan administrasi yang akan dibuat dapat membantu pihak perusahaan dalam melakukan proses layanan administrasi perawatan sepeda motor.

#### 3.2.1 *System Flow Pelayanan Administrai Perawatan Sepeda Motor*

*System flow* adalah penggambaran aliran dokumen dalam sistem dan merupakan proses kerja dalam sistem. *System flow* ini juga adalah aliran data lanjutan dari *document flow*. Jika *document flow* menggambarkan aliran data secara manual atau yang selama ini terjadi pada organisasi, maka *system flow* ini menggambarkan aliran data pada sistem yang nantinya akan dibangun untuk

membantu proses dalam organisasi. Terdapat beberapa *System flow* yang terjadi di PT. Surapita Unitrans antara lain :

1. *System flow* data *user* (pelanggan baru).
2. *System flow* jenis kendaraan.
3. *System flow* data kendaraan.
4. *System flow* jenis *service*.
5. *System flow* data *mechanic*.
6. *System flow* data *sparepart*.
7. *System flow* data hari.
8. *System flow* data perkiraan waktu.
9. *System flow* detail *user*.
10. *System flow* pendaftaran *booking service*.
11. *System flow* transaksi *administrasi*.

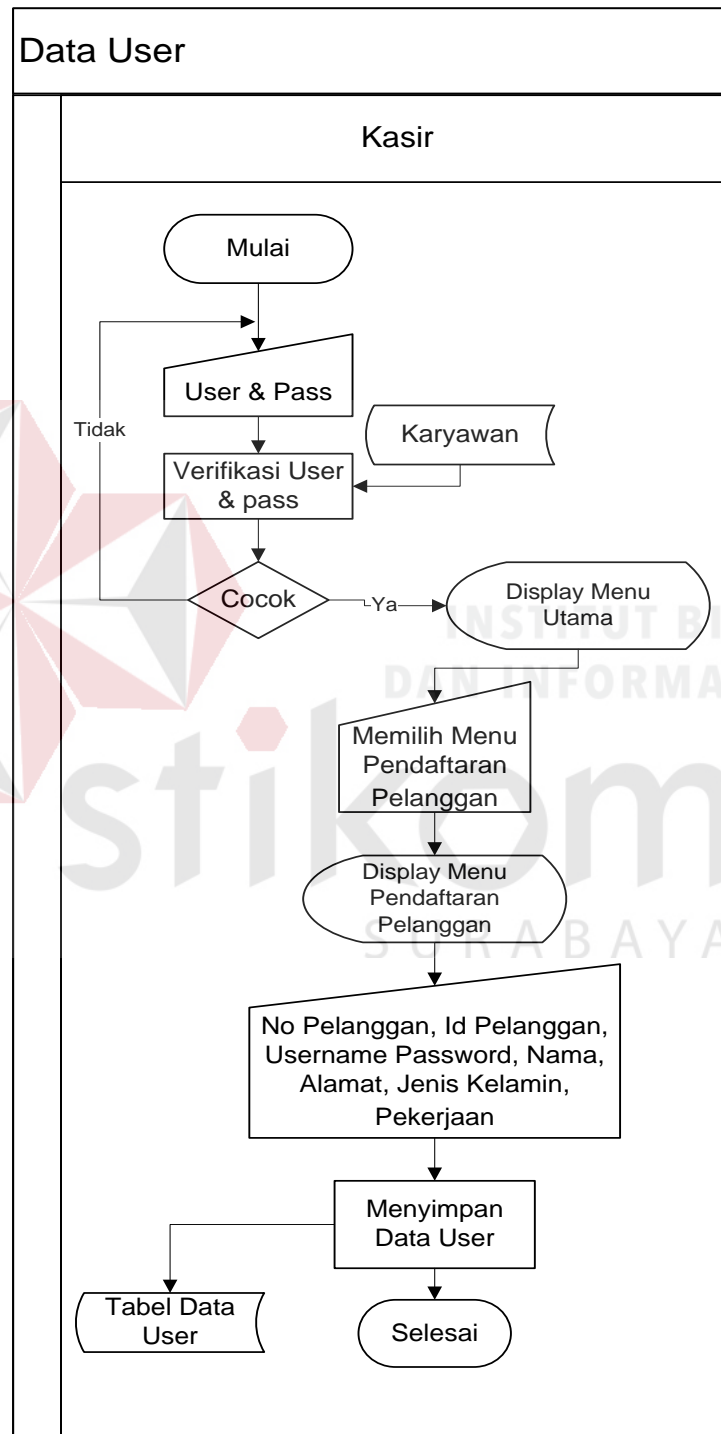
Masing-masing *system flow* akan digambarkan dan dijelaskan pada sub bab berikut ini :

#### **1. *System flow* data *user* (pelanggan baru).**

Proses pertama pada *system flow* ini adalah *system flow* pendaftaran pelanggan baru yang akan terdapat pada Gambar 3.2. Pada proses ini terdapat satu entitas yaitu admin.

Proses pertama admin pembantu proses pendaftaran pelanggan, dimana admin (*customer service*) melakukan *login* terlebih dahulu dalam proses. Setelah *login* akan terdapat *display* menu utama. Dari *display* menu utama admin dapat memilih sub dari menu master yaitu menu pendaftaran. Setelah memilih menu pendaftaran akan tampil *display* menu pendaftaran, admin

mengisi data pelanggan sesuai data yang diberikan kepada pelanggan didalam sistem tersebut. Sesudah pelanggan mengisi data maka admin menyimpan data user tersebut ke dalam tabel data user.



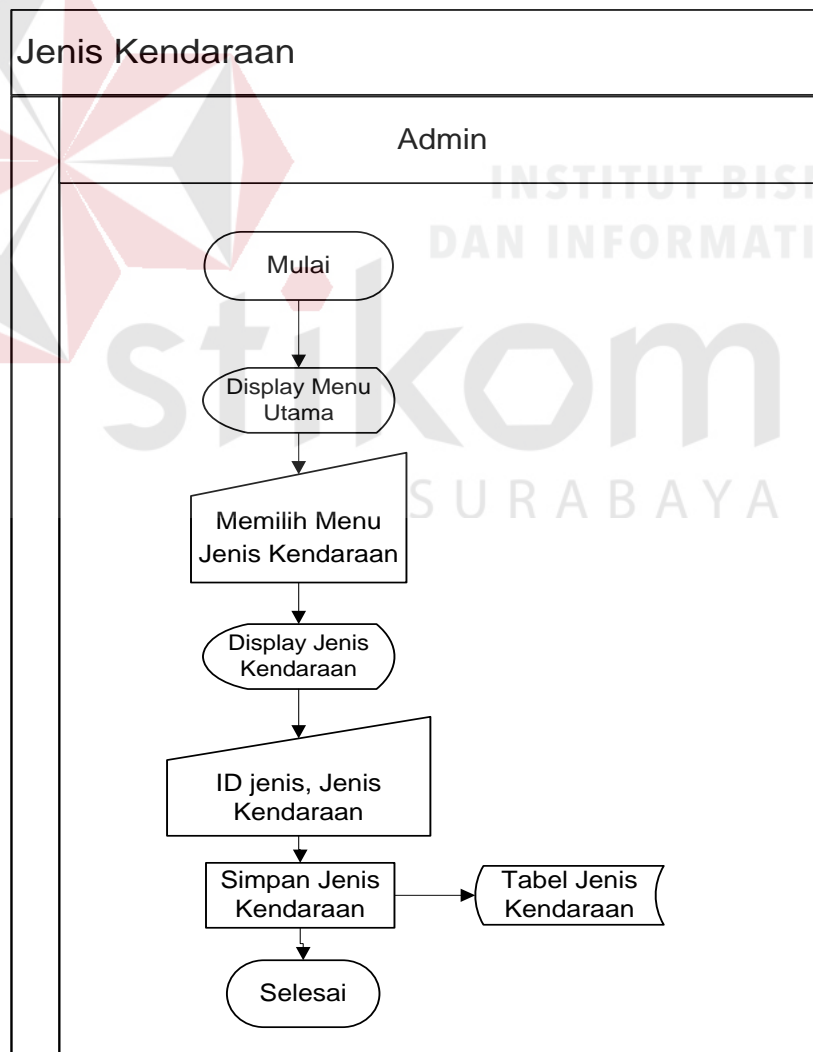
**Gambar 3.2** System Flow Pendaftaran Pelanggan (User) Baru

## 2. System flow jenis kendaraan.

System flow jenis kendaraan yang dapat ditunjukkan pada Gambar 3.3.

Diproses ini menjelaskan bahwa terdapat daftar jenis kendaraan yang dibutuhkan. Pada proses ini terdapat satu entitas yaitu admin.

Pada proses ini admin yang telah masuk dalam *display* menu utama dapat memilih menu jenis kendaraan, setelah memilih menu jenis kendaraan akan terdapat *display* menu jenis kendaraan. Admin dapat melakukan input id jenis, jenis kendaraan dan menyimpan data jenis kendaraan pada tabel jenis kendaraan.



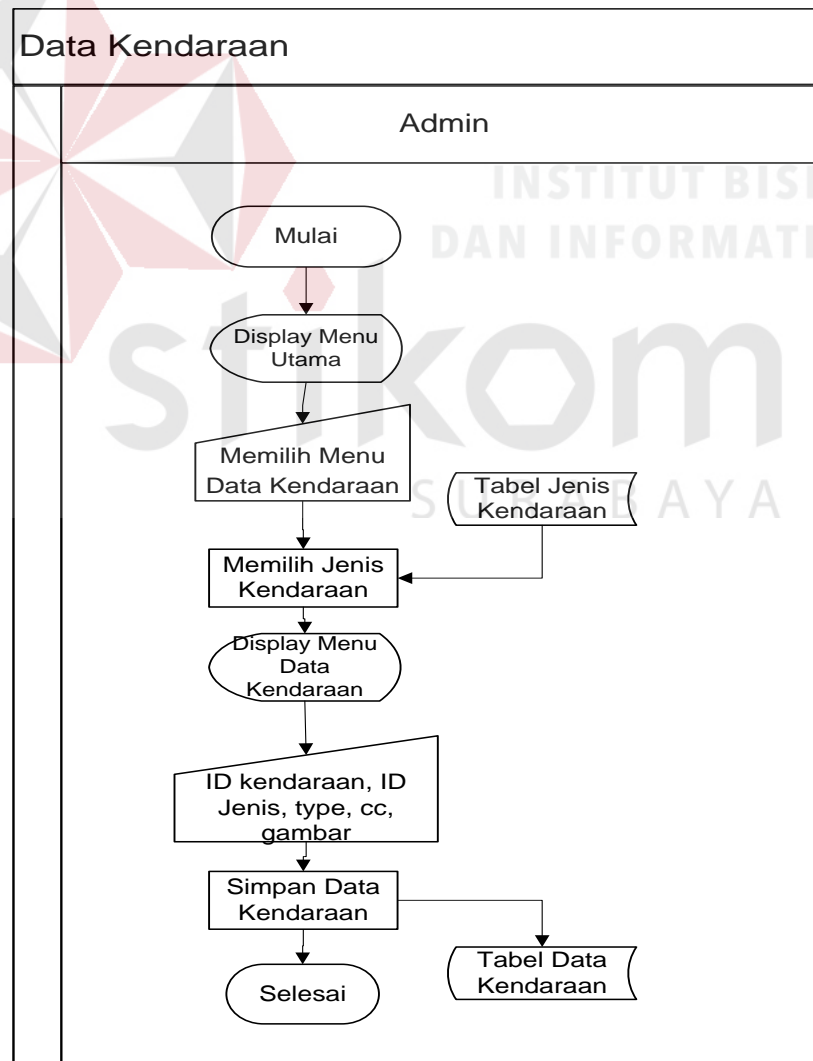
**Gambar 3.3** System Flow Jenis Kendaraan.

### 3. System flow data kendaraan.

*System flow* data kendaraan terdapat pada Gambar 3.4. Diproses ini menjelaskan bahwa terdapat daftar master data kendaraan yang dibutuhkan.

Pada proses ini terdapat satu entitas yaitu admin.

Pada proses ini admin yang telah melakukan *login* dan masuk dalam *display* menu utama, admin dapat memilih menu data kendaraan, setelah memilih menu data kendaraan akan terdapat *display* menu data kendaraan. Admin dapat melakukan input id kendaraan, id jenis kendaraan, *type*, cc, gambar dan menyimpan data kendaraan pada tabel data kendaraan.

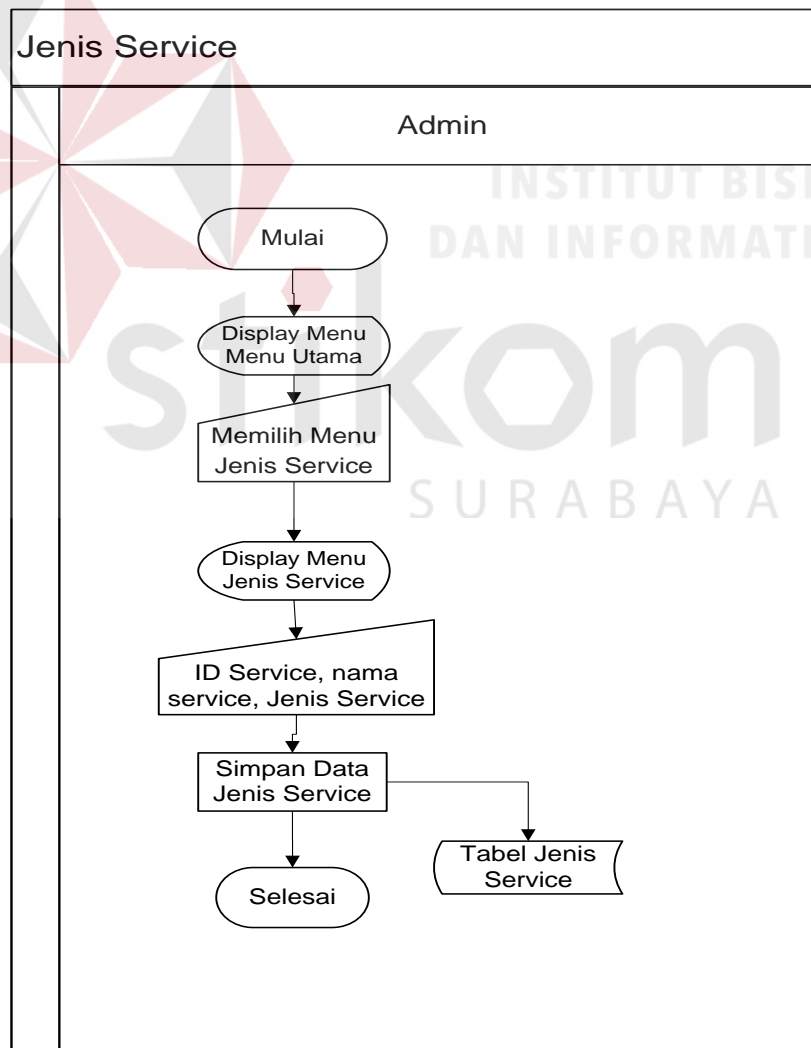


**Gambar 3.4** System Flow Data Kendaraan.



#### 4. *System flow jenis service.*

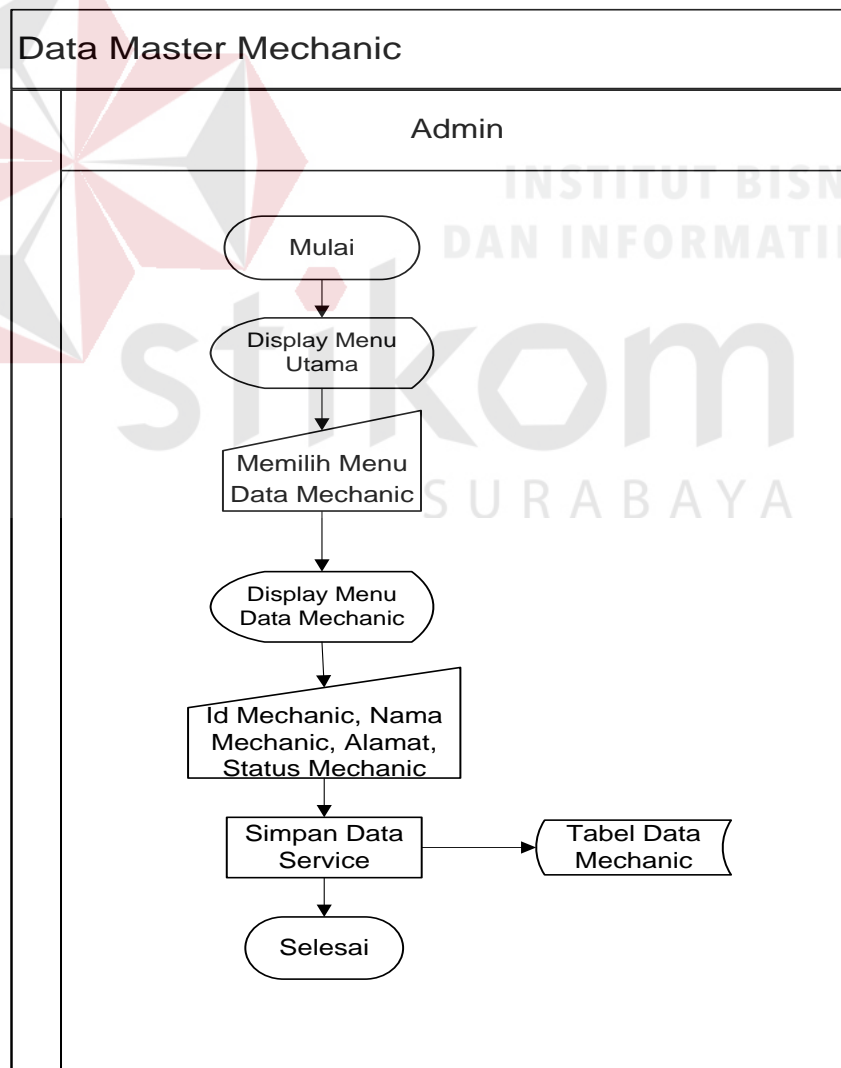
*System flow jenis service* terdapat pada Gambar 3.5. Diproses ini menjelaskan bahwa terdapat daftar *jenis service* yang dibutuhkan. Pada proses ini terdapat satu entitas yaitu admin. Pada proses ini admin yang telah *login* dan masuk dalam *display* menu utama, admin dapat memilih menu *jenis service*, setelah memilih menu *jenis service* akan terdapat *display* menu *jenis service*. Admin dapat melakukan input *id service*, *jenis service* dan menyimpan data *jenis service* pada tabel *jenis service*.



**Gambar 3.5** *System Flow Jenis Service*

### 5. System flow data mechanic.

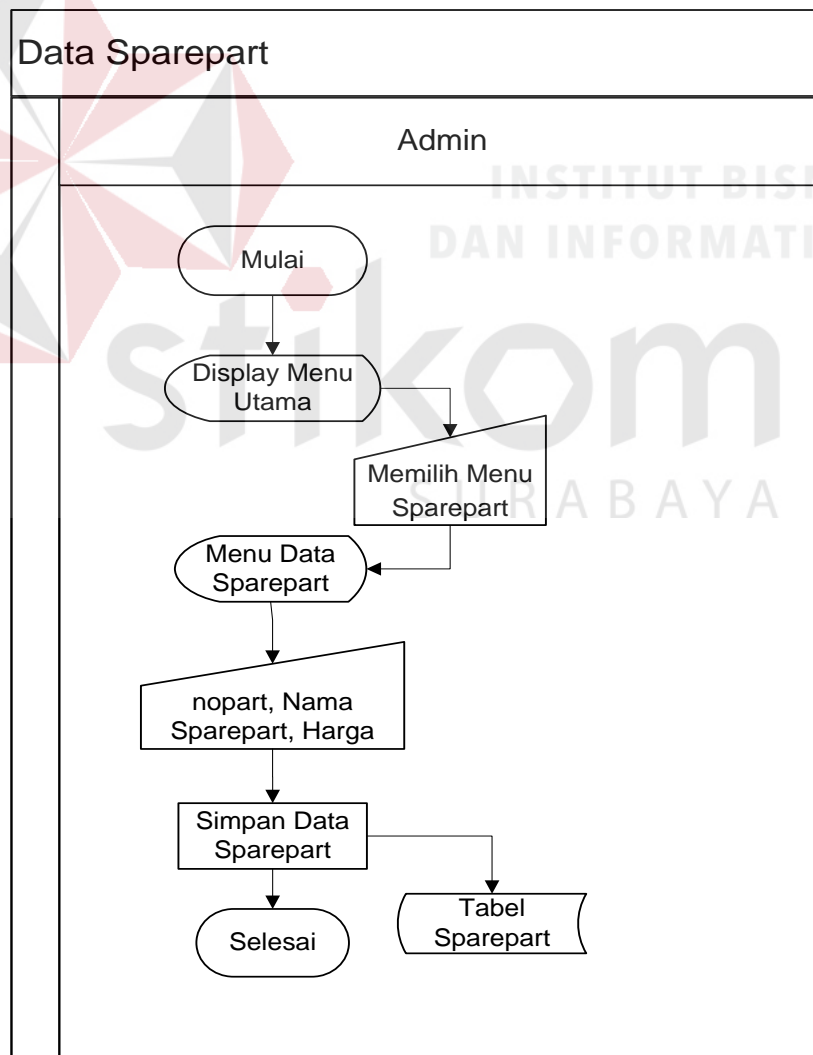
Pada *System flow data mechanic* terdapat pada Gambar 3.6. Diproses ini menjelaskan bahwa terdapat daftar data *mechani*. Pada proses ini terdapat satu entitas yaitu admin. Pada proses ini admin yang telah *login* dan masuk dalam *display* menu utama, admin dapat memilih menu data *mechanic*, setelah memilih menu data *mechanic* akan terdapat *display* menu data *mechanic*. Admin dapat melakukan input id *mechanic*, nama *mechanic*, Alamat, Status *Mechanic* dan menyimpan data *mechanic* pada tabel data *mechanic*.



**Gambar 3.6** System Flow Data Mechanic.

## 6. *System flow data sparepart.*

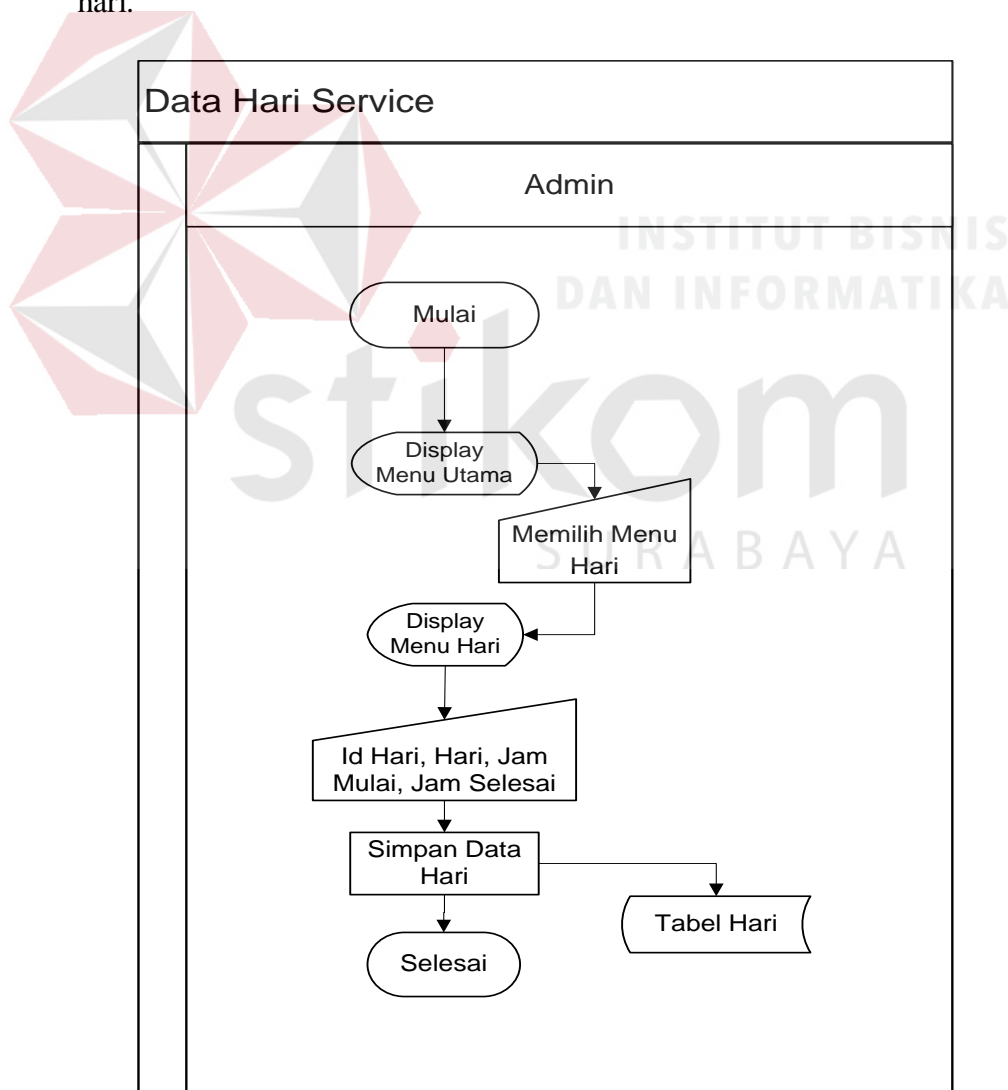
Pada *System flow data sparepart* terdapat pada Gambar 3.7. Diproses ini menjelaskan bahwa terdapat daftar data *sparepart* kendaraan yang dibutuhkan dalam perawatan sepeda motor (*service*). Pada proses ini terdapat satu entitas yaitu admin. Pada proses ini admin yang telah *login* dan masuk dalam *display* menu utama, admin dapat memilih menu data *sparepart*, setelah memilih menu data *sparepart* akan terdapat *display* menu data *sparepart*. Admin dapat melakukan input id *sparepart*, nama *sparepart*, harga dan menyimpan data *sparepart* pada tabel data *sparepart*.



**Gambar 3.7** *System Flow Data Sparepart.*

### 7. System flow data hari.

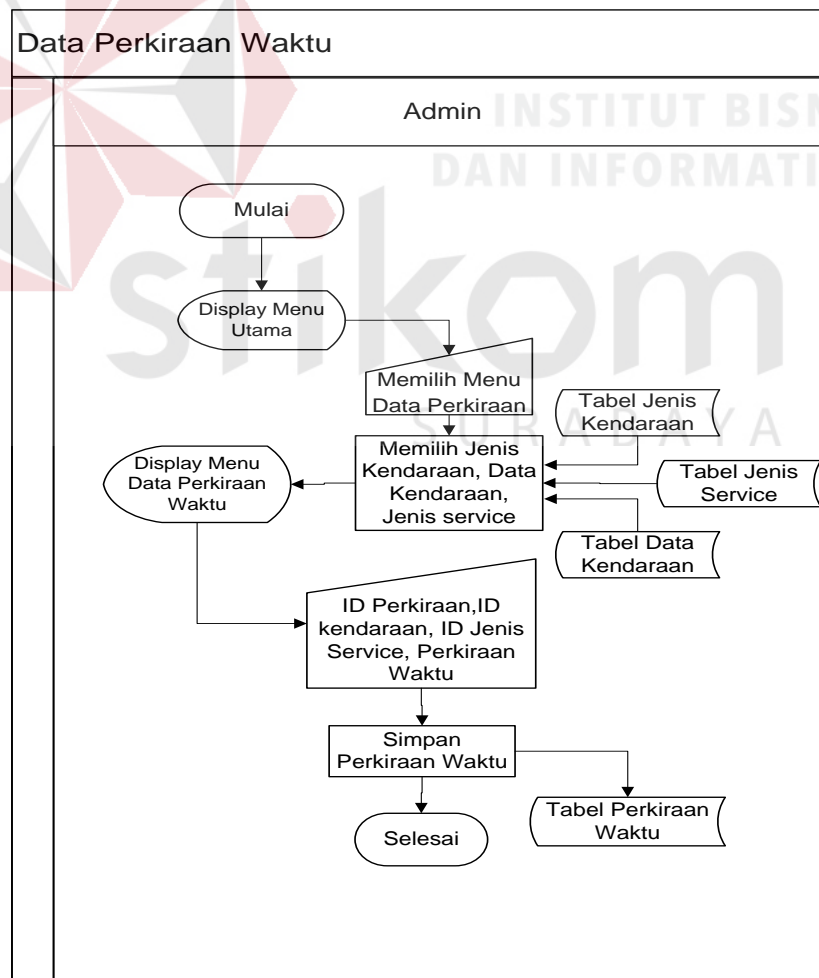
Pada *System flow* data hari terdapat pada Gambar 3.8. Diproses ini menjelaskan bahwa terdapat daftar data hari yang operasional dalam penjadwalan perawatan sepeda motor ( *service* ). Pada proses ini terdapat satu entitas yaitu admin. Pada proses ini admin yang telah *login* dan masuk dalam *display* menu utama, admin dapat memilih menu data hari, setelah memilih menu data hari akan terdapat *display* menu data hari. Admin dapat melakukan input id hari, hari, jam dan menyimpan data hari pada tabel data hari.



**Gambar 3.8** System Flow Data hari.

### 8. System flow data perkiraan waktu.

*System flow* data perkiraan waktu terdapat pada Gambar 3.9. Diproses ini menjelaskan bahwa setiap jenis *service* dan *type* kendaraan mempunyai waktu pengerjaan *service* yang berbeda-beda dalam melakukan perawatan sepeda motor. Pada proses ini terdapat satu entitas yaitu admin. Pada proses ini admin yang telah *login* dan masuk dalam *display* menu utama, admin dapat memilih menu data perkiraan waktu, setelah memilih menu data perkiraan waktu akan terdapat *display* menu data perkiraan waktu. Admin dapat melakukan input id perkiraan, id kendaraan, id jenis, perkiraan waktu dan menyimpan data perkiraan waktu pada tabel data perkiraan.

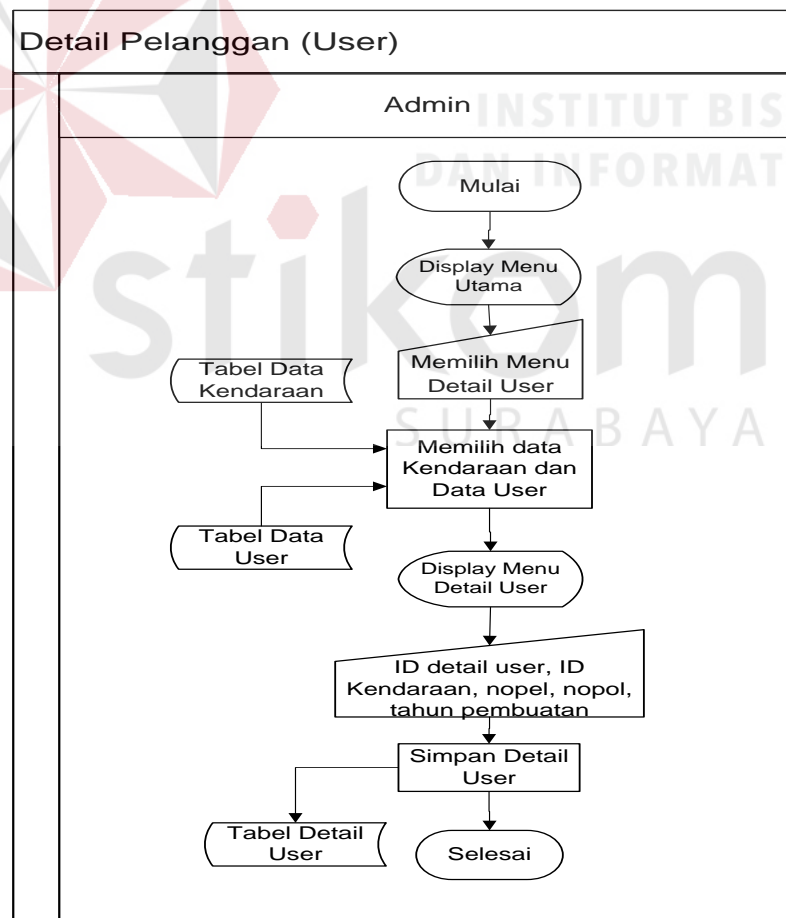


**Gambar 3.9** System Flow Detail Perkiraan Waktu.

### 9. System flow detail user.

*System flow* detail *user* (Detail Pelanggan) terdapat pada Gambar 3.10. Diproses ini menjelaskan bahwa data pelanggan mempunyai detail pelanggan, dimana satu pelanggan bisa memiliki beberapa macam *type* kendaraan. Pada proses ini terdapat satu entitas yaitu admin.

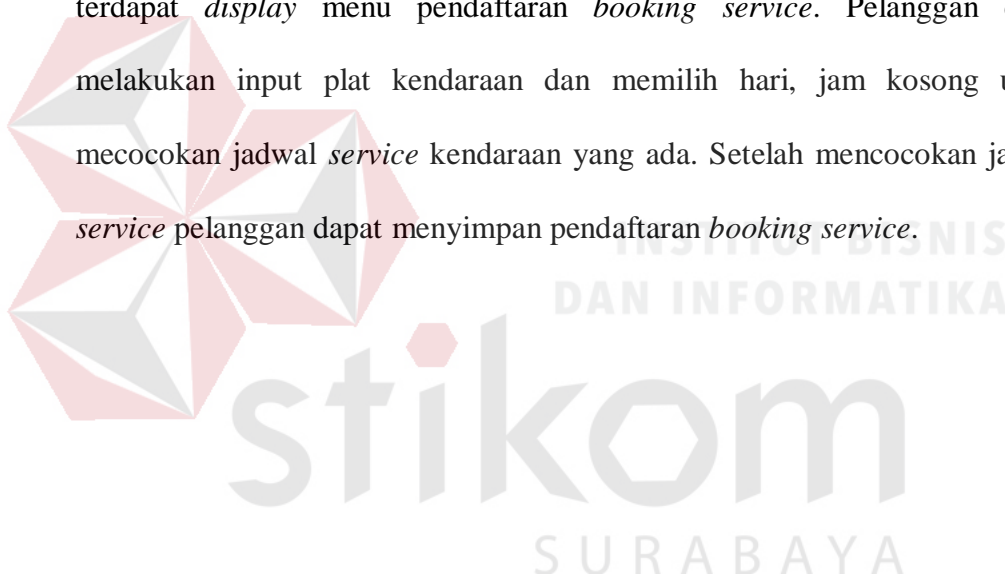
Pada proses ini admin yang telah *login* dan masuk dalam *display* menu utama, admin dapat memilih menu detail *user* akan terdapat *display* menu detail *user*. Admin dapat melakukan input id detail *user*, id jenis, nopel, nomer polisi, tahun pembuatan dan menyimpan data kendaraan pada tabel detail pelanggan (*user*).

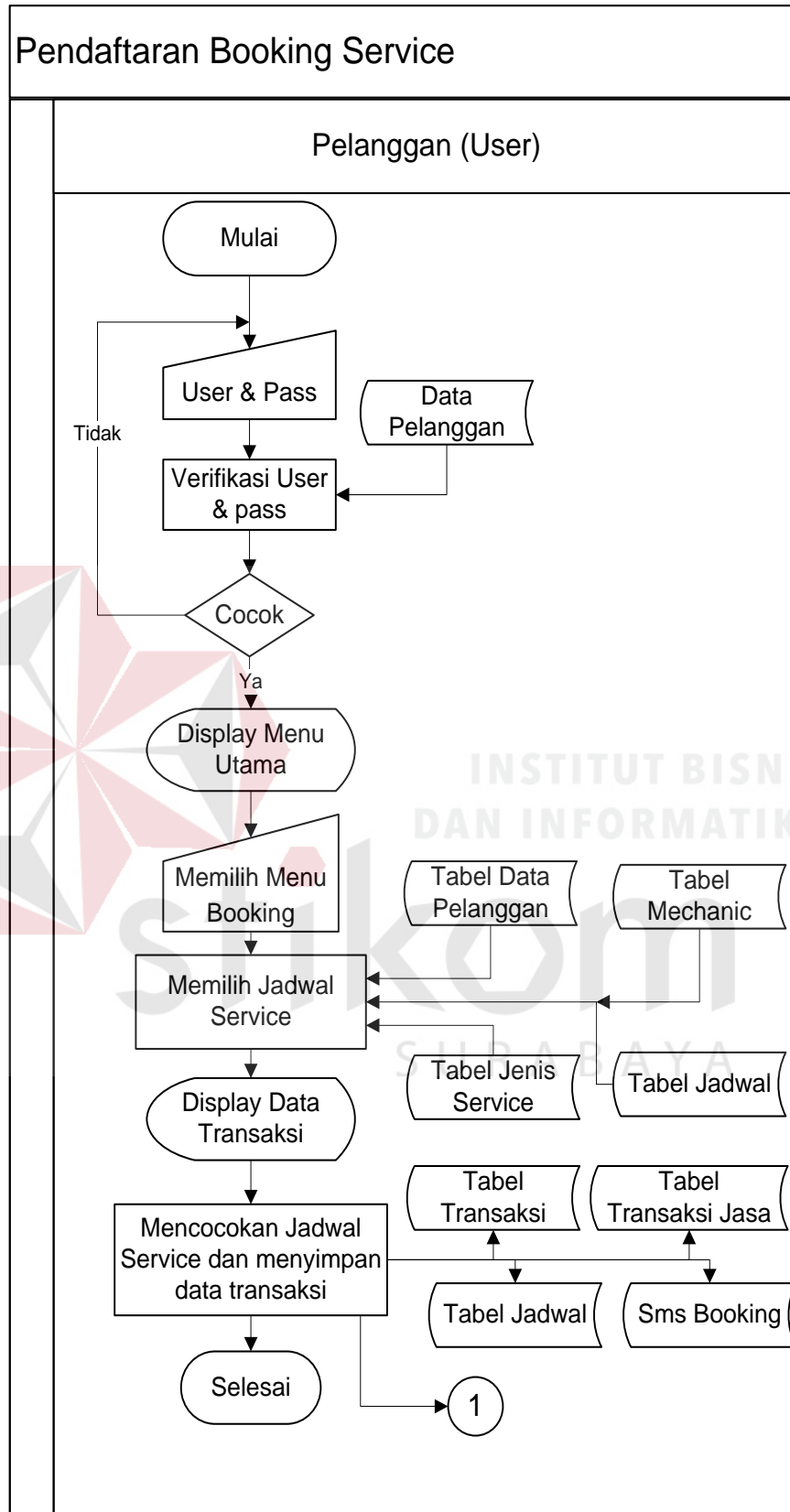


**Gambar 3.10** System Flow Detail User.

#### 10. *System flow* pendaftaran *booking service*.

*System flow* pendaftaran *booking service* terdapat pada Gambar 3.14. Diproses ini menjelaskan alur sistem dalam proses *booking* perawatan sepeda motor. Pada proses ini terdapat satu entitas yaitu pelanggan. Dimana pelanggan dapat mengakses *website* layanan administrasi ini dengan melakukan *login* terlebih dahulu. Pelanggan yang telah *login* dan masuk dalam *display* menu utama, pelanggan dapat memilih menu pendaftaran *booking service*, setelah memilih menu pendaftaran *booking service* akan terdapat *display* menu pendaftaran *booking service*. Pelanggan dapat melakukan input plat kendaraan dan memilih hari, jam kosong untuk mencocokkan jadwal *service* kendaraan yang ada. Setelah mencocokkan jadwal *service* pelanggan dapat menyimpan pendaftaran *booking service*.





**Gambar 3.11** System Flow Pendaftaran Booking Service

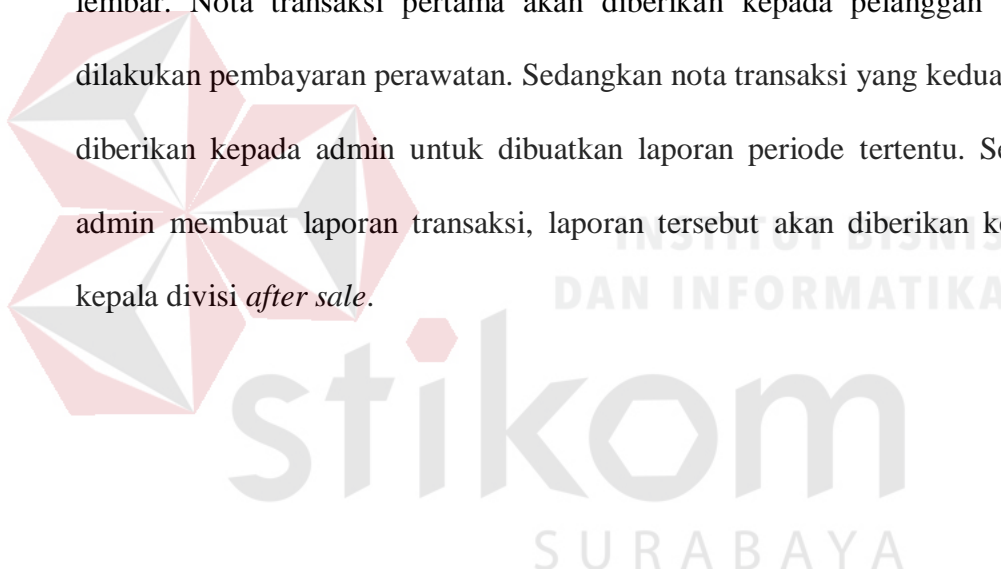


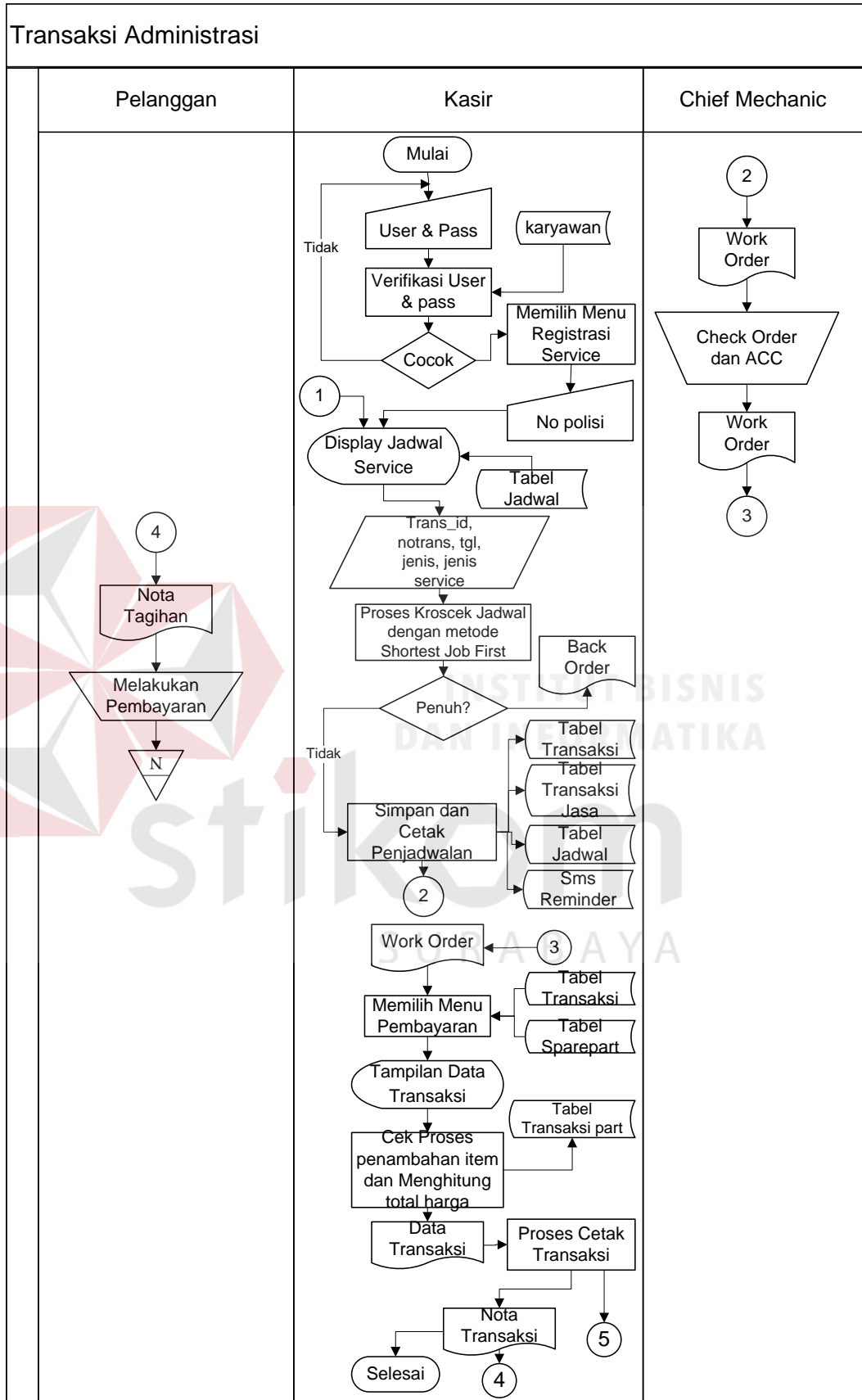
## 11. *System flow* transaksi *administrasi*.

Pada *System flow* transaksi *administrasi* terdapat pada Gambar 3.13 dan Gambar 3.14. Diproses ini menjelaskan alur sistem *administrasi* secara keseluruhan yang saling berkaitan antara satu bagian dengan bagian yang lain. Pada proses ini terdapat lima entitas yaitu pelanggan, *customer service* (admin), *chief mechanic*, kasir, dan kepala divisi *after sale* .

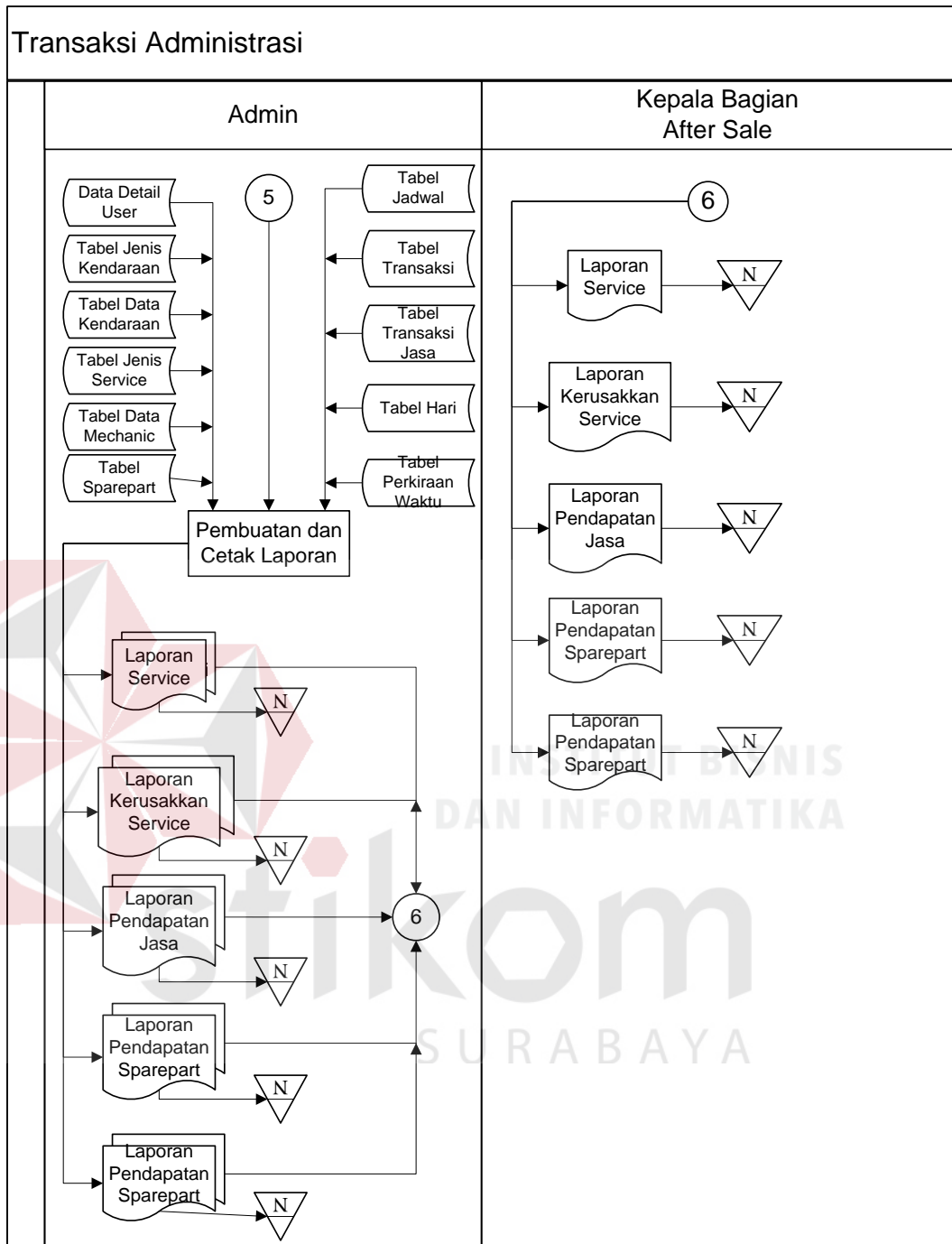
Pada bagian ini kasir akan *login* terlebih dahulu. Setelah *login* admin memilih menu registrasi *service*, dimana pelanggan yang melakukan *service* akan memasukan no polisi terlebih dahulu ke sistem. No polisi tersebut akan di *check* apakah sudah terdaftar sebagai pelanggan apa tidak. Jika terdaftar akan menampilkan data *user* (pelanggan) dan dapat memilih jenis *service* kendaraan. setelah memilih jenis *service* sistem akan melakukan proses kroscek jadwal apakah jadwal yang terjadi pada hari tersebut penuh atau tidak. Jika jadwal penuh order kembali. Sedangkan jika pada hari tersebut jadwal tidak penuh admin dapat langsung menyimpan dan mencetak penjadwalan pada hari tersebut untuk dijadikan *work order*. Selanjutnya *work order* akan diserahkan kepada *chief mechanic*, dan *chief mechanic* memberikan *work order* kepada *mechanic* untuk melakukan perawatan sepeda motor (*service*). Setelah perawatan (*service*) selesai, *mechanic* akan men-*check* keadaan sepeda motor apakah ada pergantian *sparepart* atau tidak. Jika tidak ada *mechanic* dapat langsung melakukan perawatan sepeda motor. sedangkan jika ada pergantian *sparepart*, *mechanic* akan meminta persetujuan kepada pelanggan apakah diganti dengan yang baru atau tidak. Setelah perawatan selesai *mechanic* memberikan laporan pengerjaan kepada

*chief mechanic*. Selanjutnya *chief mechanic* melakukan *check* laporan pengerjaan. Setelah selesai *chief mechanic* akan menyerahkan laporan tersebut kepada kasir, dan kasir akan menghitung total biaya perawatan (*service*) yang telah dilakukan untuk menjadi data transaksi. Data transaksi tersebut akan di-*check* kembali oleh petugas admin apakah sesuai atau tidak dengan work order dari *chief mechanic*, jika tidak sesuai kasir akan melakukan kembali *check* penambahan *sparepart* dan menghitung total biaya. Sedangkan jika sesuai kasir dapat mencetak nota tagihan sebanyak 2 lembar. Nota transaksi pertama akan diberikan kepada pelanggan untuk dilakukan pembayaran perawatan. Sedangkan nota transaksi yang kedua akan diberikan kepada admin untuk dibuatkan laporan periode tertentu. Setelah admin membuat laporan transaksi, laporan tersebut akan diberikan kepada kepala divisi *after sale*.





Gambar 3.12 System Flow Transaksi Administrasi.

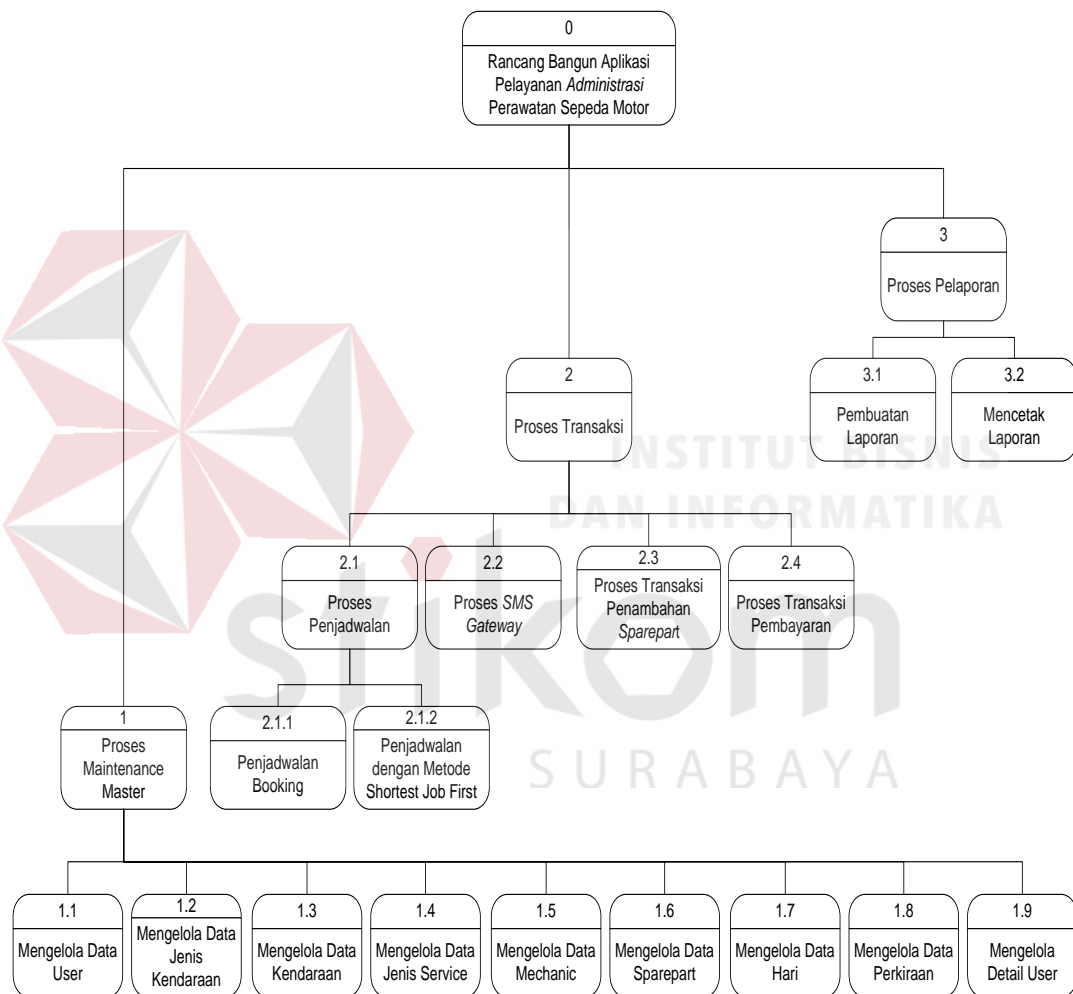


**Gambar 3.13** System Flow Transaksi Administrasi.

### 3.2.2 Diagram HIPO

Berdasarkan system flow yang telah dibuat diatas maka dapat kembangkan kedalam diagram HIPO. Diagram ini menggambarkan hubungan dari modul-modul dalam suatu sistem secara berjenjang. Selain itu diagram HIPO juga

menunjukkan secara garis besar hubungan dari input, proses dan output, dimana bagian input menunjukkan item-item data yang akan digunakan oleh bagian proses yang berisi langkah-langkah kerja dari fungsi atau modul dan bagian output berisi hasil pemrosesan data. Adapun penggambaran diagram *HIPO* dalam sistem ini adalah sebagai berikut :



**Gambar 3.14** Diagram *HIPO*

Diagram *HIPO* diatas menunjukkan tiga proses utama dalam sistem yaitu : pertama, mengelola data master, berguna untuk *input* atau *update* data master yang nantinya akan digunakan dalam transaksi, data master tersebut antara lain data pelanggan, data *mechanic*, data *sparepart*, data jadwal *service*, data

kendaraan, jenis kendaraan, dan data jenis *service*. Kedua, mengelola transaksi, berguna untuk proses pengelolaan pendaftaran pelanggan baru, transaksi penjadwalan dengan metode *shortest job first*, pembelian sparepart, dan transaksi pembayaran. Ketiga, membuat laporan, berguna untuk dokumentasi sistem terhadap report-report yang diperlukan, antara lain: memilih laporan, mencetak laporan.

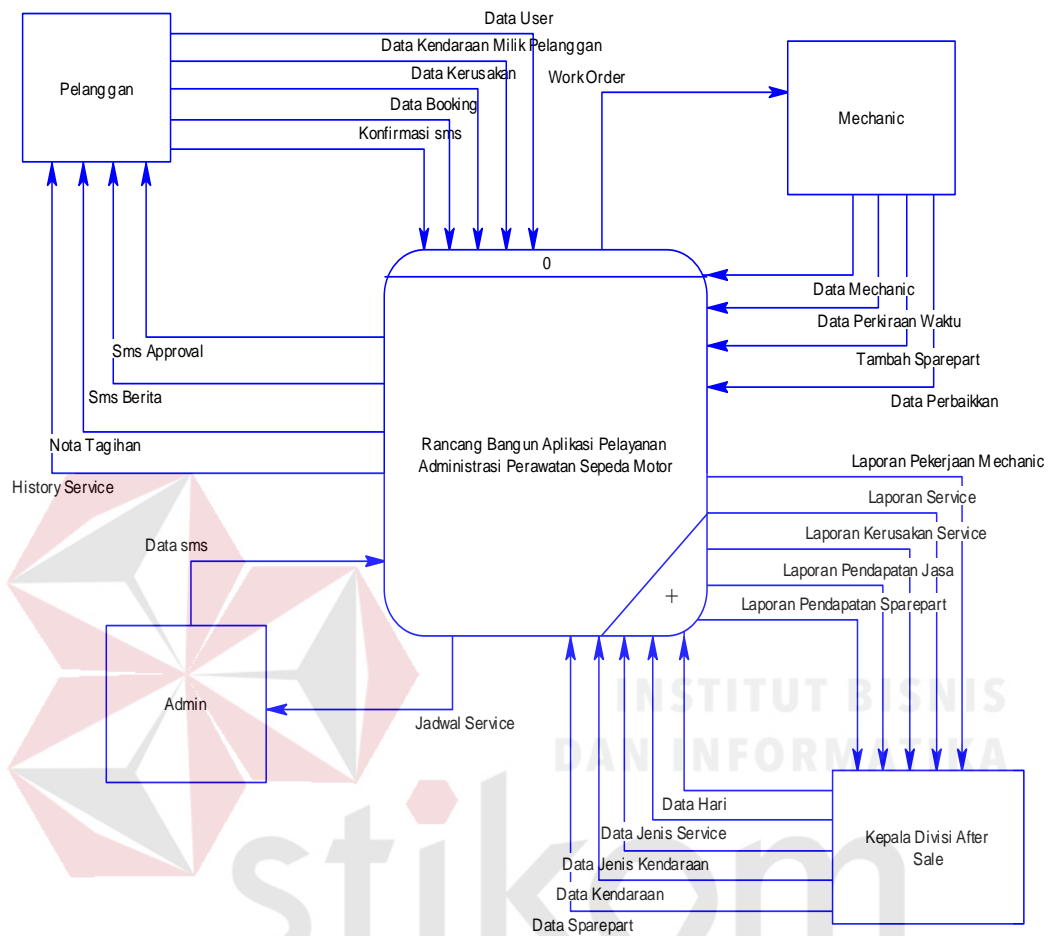
### 3.2.3 Data Flow Diagram

*Data Flow Diagram (DFD)* merupakan gambaran aliran data yang terdapat dalam sistem. Diagram ini menjelaskan secara lebih detail tentang proses yang terdapat pada diagram *HIPO* dengan alur data yang terjadi pada setiap prosesnya masing-masing. *DFD* ini berfungsi untuk menggambarkan proses aliran data yang terjadi di dalam sistem mulai dari yang paling tinggi sampai yang paling rendah, sehingga nantinya akan dimungkinkan proses dekomposisi, partisi, atau pembagian sistem ke dalam bagian-bagian yang lebih kecil dan lebih sederhana. Adapun penjelasan dari *DFD* tersebut dapat dilihat sebagai berikut :

#### A. Context Diagram

*Context Diagram* merupakan diagram pertama dalam rangkaian *DFD* yang menunjukkan entitas-entitas yang berhubungan dengan sistem. Diagram ini juga akan menggambarkan secara umum tentang *input-output* ke dalam sistem. *Context Diagram* rancang bangun aplikasi pelayanan administrasi perawatan sepeda motor di PT. Surapita Unitrans Surabaya ini terdapat empat entitas, yaitu : pelanggan,

admin (*customer service*), kepala divisi *after sale*, kasir sebagaimana ditunjukkan pada gambar berikut :



**Gambar 3.15** Context Diagram

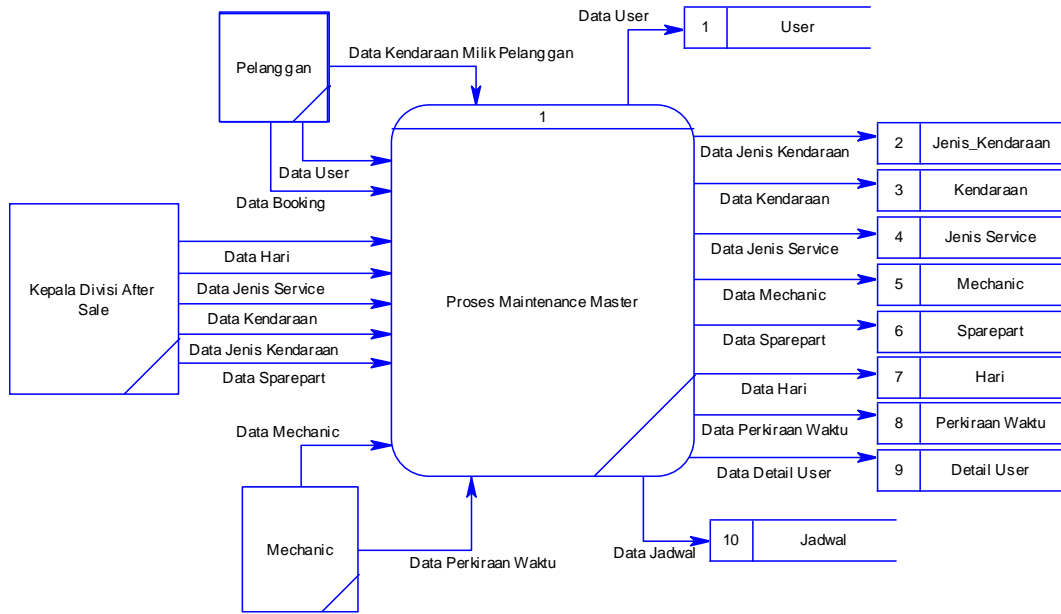
Pada gambar *context diagram* diatas menunjukkan aliran pertama dilakukan oleh pelanggan yang menginputkan data pelanggan, data kendaraan milik pelanggan, data kerusakan, data booking, dan konfirmasi sms. Selanjutnya sistem akan mengolahnya dan menghasilkan *output* berupa nota tagihan, *sms approval*, *sms* berita dan *history service*. Disamping itu, *input output* yang juga terjadi pada entitas admin dalam penggambaran *context diagram* ini adalah *input*: data *sms*, sedangkan *output*nya adalah *jadwal service*, untuk bagian kepala divisi *after sale* *input*nya adalah data *mechanic*, data *sparepart*, data hari, data kendaraan, data jenis kendaraan, data jenis *service*, dan persetujuan laporan.

Sedangkan *outputnya* adalah laporan data master, laporan *service*, laporan kerusakan *service*, laporan pendapatan jasa, laporan pendapatan sparepart. *Mechanic* memiliki *input* tambah *sparepart*, data perbaiki, data perkiraan waktu, data *mechanic* yang menghasilkan *output work order*.

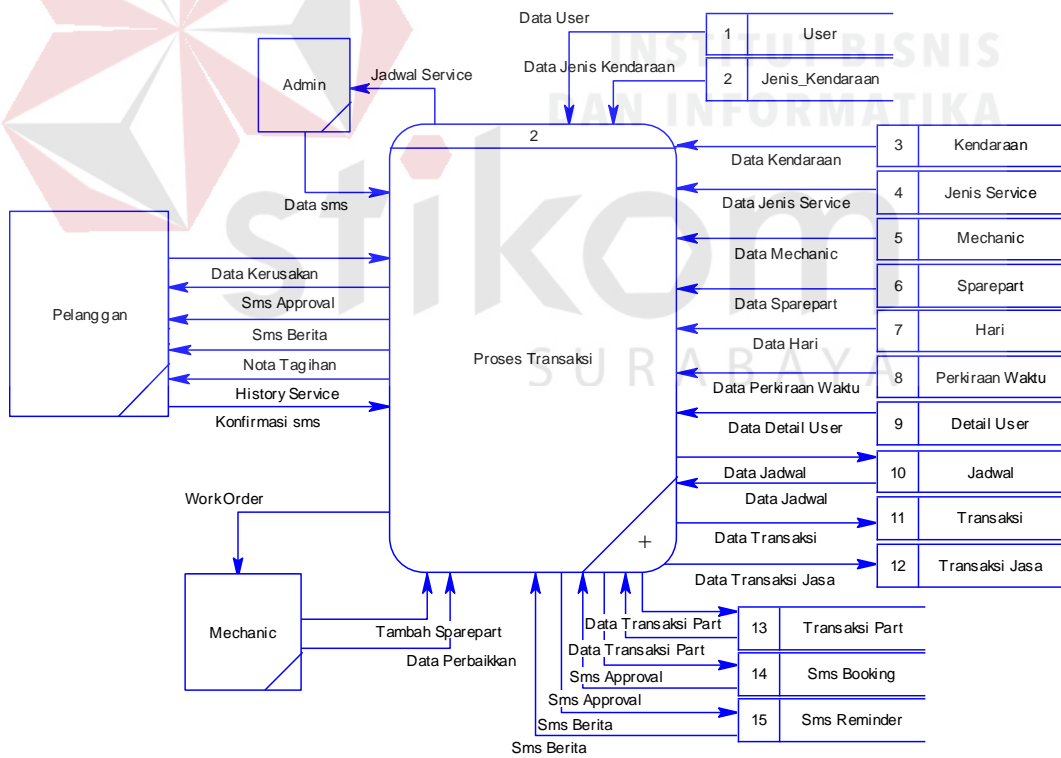
## **B. DFD Level 0**

*DFD Level 0* merupakan hasil *decompose* dari *context diagram*, yang mana menjelaskan lebih rinci tiap aliran data dan proses-proses didalamnya. Tiap proses tersebut akan membuat hubungan yang saling terkait satu sama lain sehingga membentuk aliran proses yang menggambarkan proses rancang bangun aplikasi pelayanan *administrasi* perawatan sepeda motor di PT. Surapita Unitrans Surabaya. Pada *DFD Level 0* ini terdapat tiga proses utama, antara lain : mengelola *maintenance data master*, mengelola transaksi, dan membuat laporan. Selain itu terdapat beberapa *data store* yang berguna menyimpan data hasil proses tiap fungsi yaitu : data jenis *service*, jenis kendaraan, kendaraan, perkiraan waktu, *mechanic*, hari, *sparepart*, *sms*, data transaksi, data detail transaksi dan data pelanggan . Untuk lebih jelasnya dapat dilihat pada gambar *DFD Level 0* berikut ini:

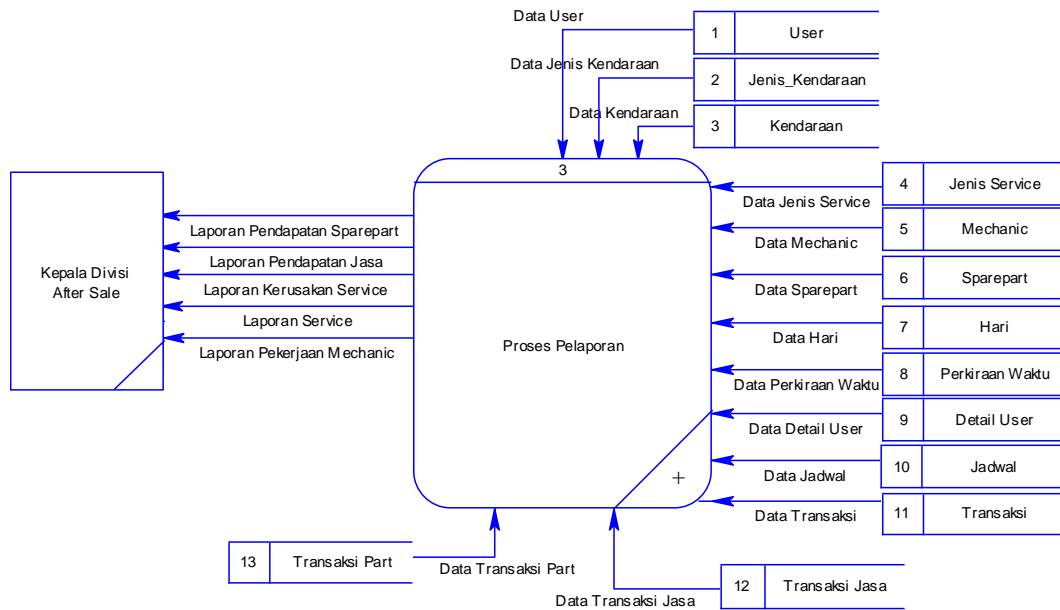




Gambar 3.16 DFD level 0 Melakukan Proses Maintenance Master



Gambar 3.17 DFD level 0 Melakukan Proses Transaksi

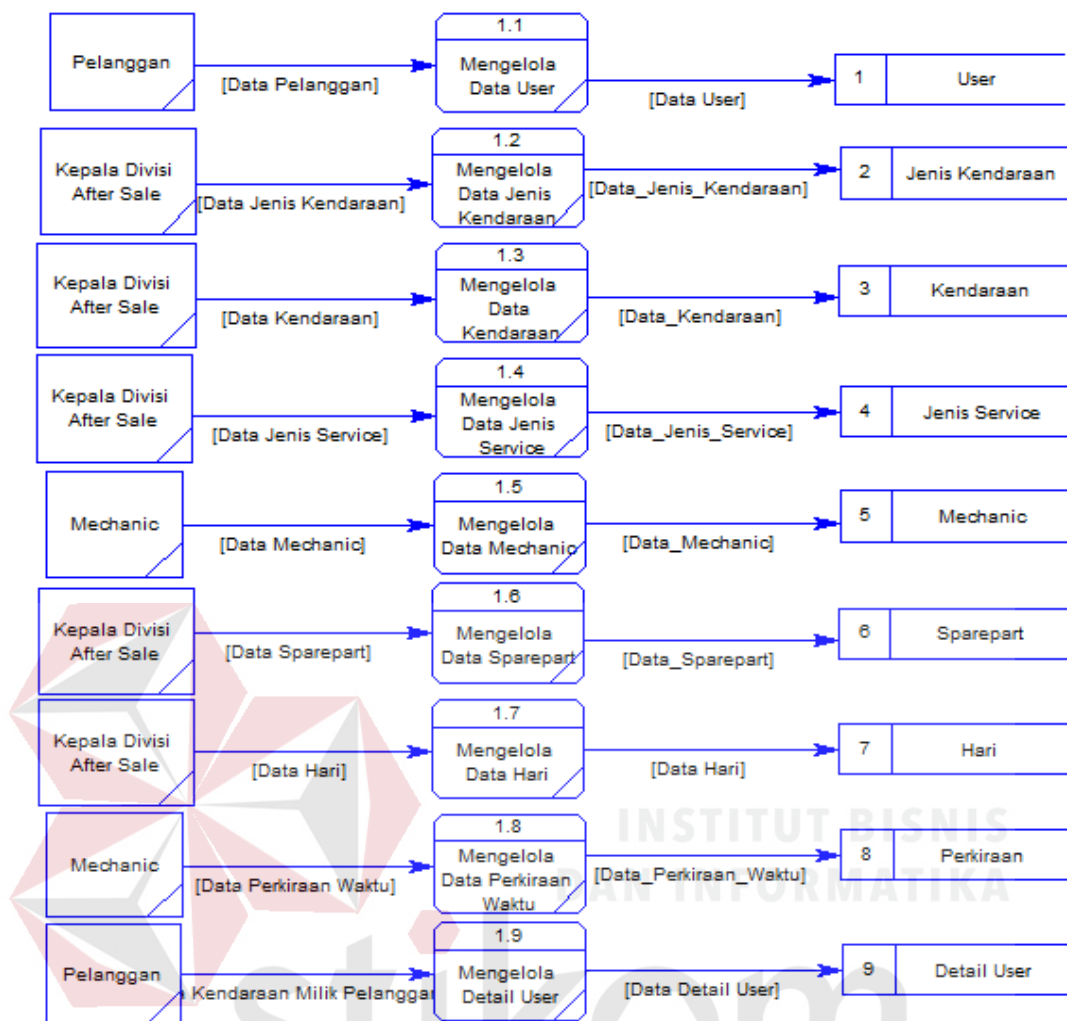


**Gambar 3.18** DFD level 0 Melakukan Proses Pelaporan

Seperti penjelasan *DFD Level 0* diatas, bahwa gambar 3.17, gambar 3.18, dan gambar 3.19 ini memiliki tiga proses dan dua belas *data store* yang fungsinya masing-masing adalah penjabaran lebih lanjut tentang proses dalam sistem dan tabel yang digunakan dalam penyimpanan data. Selanjutnya, tiga proses tersebut juga dijelaskan lebih detail kedalam *DFD Level 1* berikut :

### C. DFD Level 1 Melakukan Maintenance Master

Pada *DFD Level 1* mengelola data master ini adalah *decompose* atau proses *breakdown* dari mengelola data master yang terdapat pada *DFD Level 0*. Proses ini menjelaskan *input* data baru ataupun *update* data yang sudah ada sehingga akan tersusun beberapa sub proses dan *data store* yang mendukung didalamnya, sebagaimana ditunjukkan pada gambar berikut :

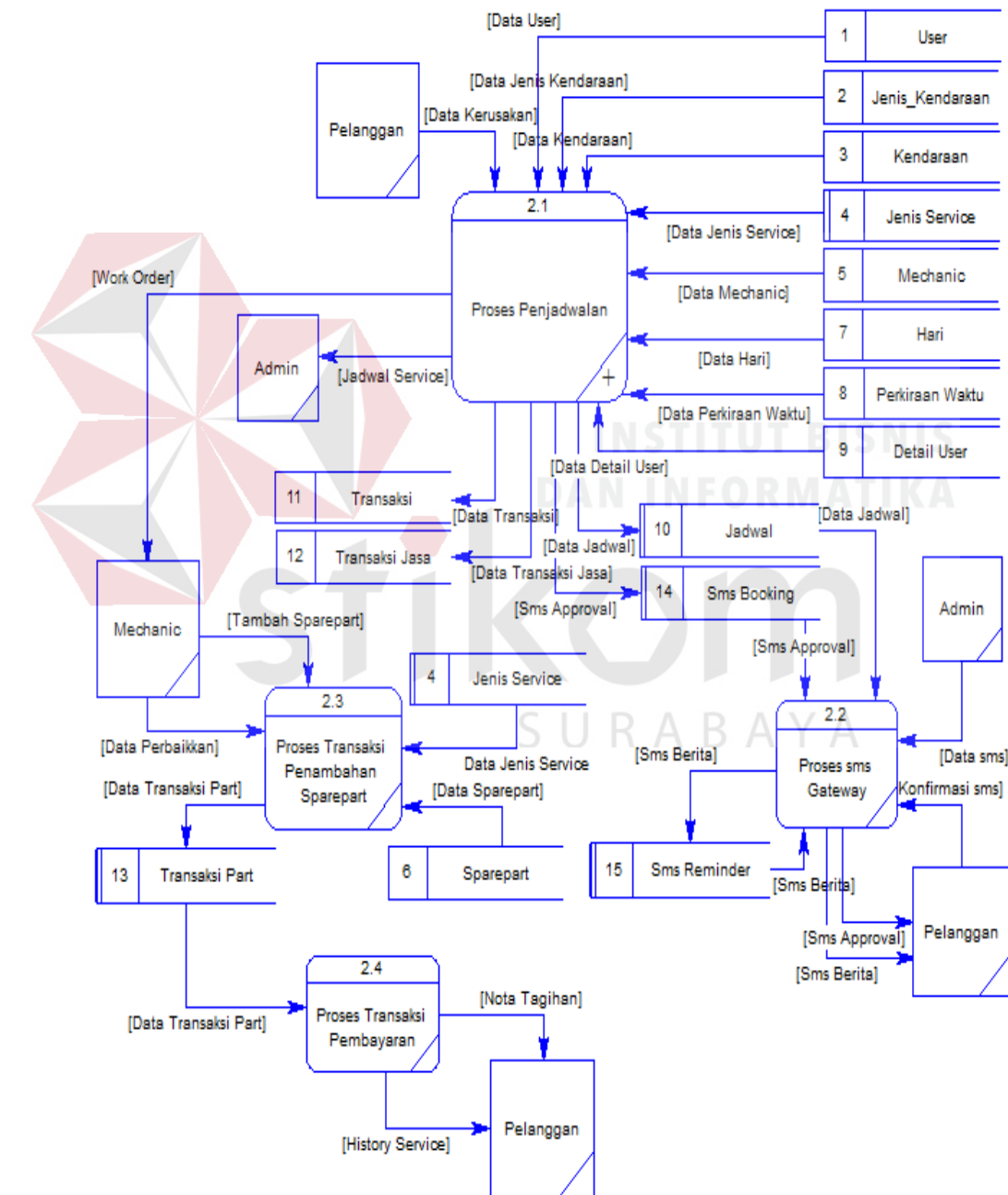


**Gambar 3.19** DFD Level 1 Mengelola Maintenance Master

Dari gambar diatas menunjukkan ada sepuluh sub proses yang berfungsi mengelola data master, yaitu: mengelola data *user*, mengelola data jenis kendaraan, mengelola data kendaraan, mengelola data jenis *service*, mengelola data *mechanic*, mengelola data *sparepart*, mengelola data hari, mengelola data perkiraan waktu, mengelola detail *user*, mengelola data jadwal. Selanjutnya masing-masing proses tersebut data disimpan kedalam data *store user*, jenis *service*, jenis kendaraan, kendaraan, perkiraan waktu, *mechanic*, hari, *sparepart*, detail *user*, jadwal *service*.

#### D. DFD Level 1 Mengelola Transaksi

DFD Level 1 selanjutnya adalah sub proses mengelola transaksi. Fungsi proses ini adalah mengelola transaksi pelayanan administrasi perawatan sepeda motor di PT.Surapita Unitrans Surabaya. Adapun penggambarannya adalah sebagai berikut :

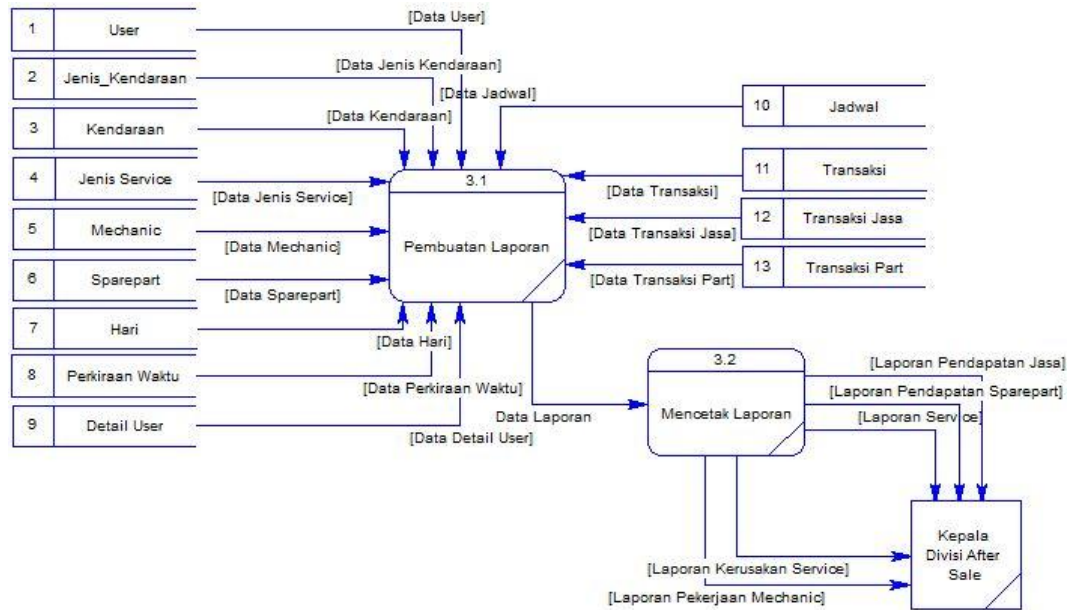


Gambar 3.20 DFD Level 1 Mengelola Transaksi

Pada proses ini pelanggan memasukkan data pelanggan di dalam sistem, setelah data pelanggan tersimpan, pelanggan dapat melakukan *service* dengan memasukkan data *service*. Dimana data kerusakan tersebut dan data booking untuk pelanggan yang melakukan booking, dan akan tersimpan sebagai data transaksi. Admin akan menerima data tersebut proses transaksi yang telah dilakukan pelanggan. Dari proses transaksi tersebut sistem akan mengirimkan *sms* berupa *sms approval* untuk jadi atau tidaknya pelanggan melakukan *service* untuk pelanggan yang melakukan *booking* atau *sms* berita sebagai *reminder*. Dari transaksi tersebut *mechanic* akan menerima *work order* pekerjaan, setelah pekerjaan selesai *work order* tersebut diberikan kesistem transaksi penambahan *sparepart* untuk melakukan *krosscek* apakah ada penambahan *sparepart* selama transaksi atau tidak. Setelah selesai melakukan cek penambahan *sparepart* maka akan dicetak nota tagihan beserta total biaya transaksi. Pelanggan yang telah melakukan *service* dan melakukan pembayaran akan menerima nota tagihan dan dapat melihat *history service* pelanggan melalui website yang telah disediakan pihak perusahaan.

#### **E. DFD Level 1 Membuat Laporan**

*DFD Level 1* membuat laporan ini adalah proses pembuatan laporan yang terkait dengan rancang bangun pelayanan administrasi perawatan sepeda moto di PT. Surapita Unitrans Surabaya. Adapun penggambarannya adalah sebagai berikut :

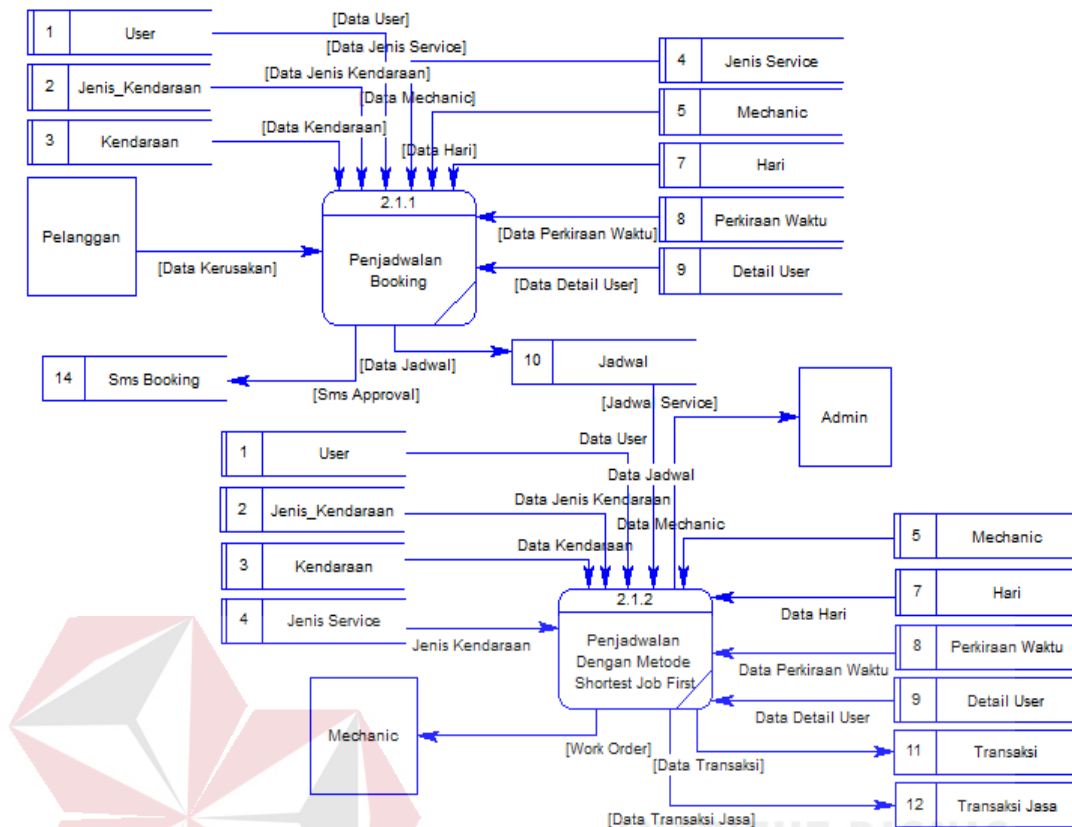


**Gambar 3.21 DFD Level 1 Membuat Laporan**

*DFD Level 1* membuat laporan ini menggambarkan aliran proses pembuatan laporan antara lain : laporan data master, laporan *service*, laporan kerusakan *service*, laporan *service*, laporan pendapatan jasa, laporan pendapatan *sparepart*. Data laporan ini diambil dari *data store user*, jenis kendaraan, kendaraan, jenis *service*, *mechanic*, *sparepart*, hari, perkiraan waktu, *detail user*, transaksi, transaksi jasa, transaksi part.

#### **F. DFD Level 2 Melakukan Proses Penjadwalan.**

*DFD Level 2* selanjutnya adalah sub proses melakukan proses penjadwalan. Fungsi proses ini adalah mengelola penjadwalan pelayanan administrasi perawatan sepeda motor di PT. Surapita Unitrans Surabaya. Adapun penggambarannya adalah sebagai berikut:



**Gambar 3.22** DFD Level 2 Melakukan Proses Penjadwalan

Pada proses ini pelanggan melakukan *login* terlebih dahulu didalam sistem agar dapat melakukan penjadwalan service, pelanggan yang sudah *login* dapat melakukan *service* dengan memasukkan data *service*. Dimana data *service* tersebut akan tersimpan sebagai jadwal *service* dan data transaksi. Admin akan menerima jadwal *service* dari proses transaksi yang telah dilakukan pelanggan yang telah melakukan proses penjadwalan booking, dari proses transaksi tersebut sistem akan menyimpan data booking tersebut. Sedangkan pelanggan yang ingin *service* dan datang langsung tanpa melakukan proses booking akan di jadwalkan oleh sistem dengan melakukan penjadwalan dengan metode *shortest job first*. Dimana jadwal tersebut menyesuaikan jadwal yang telah dipesan oleh pelanggan

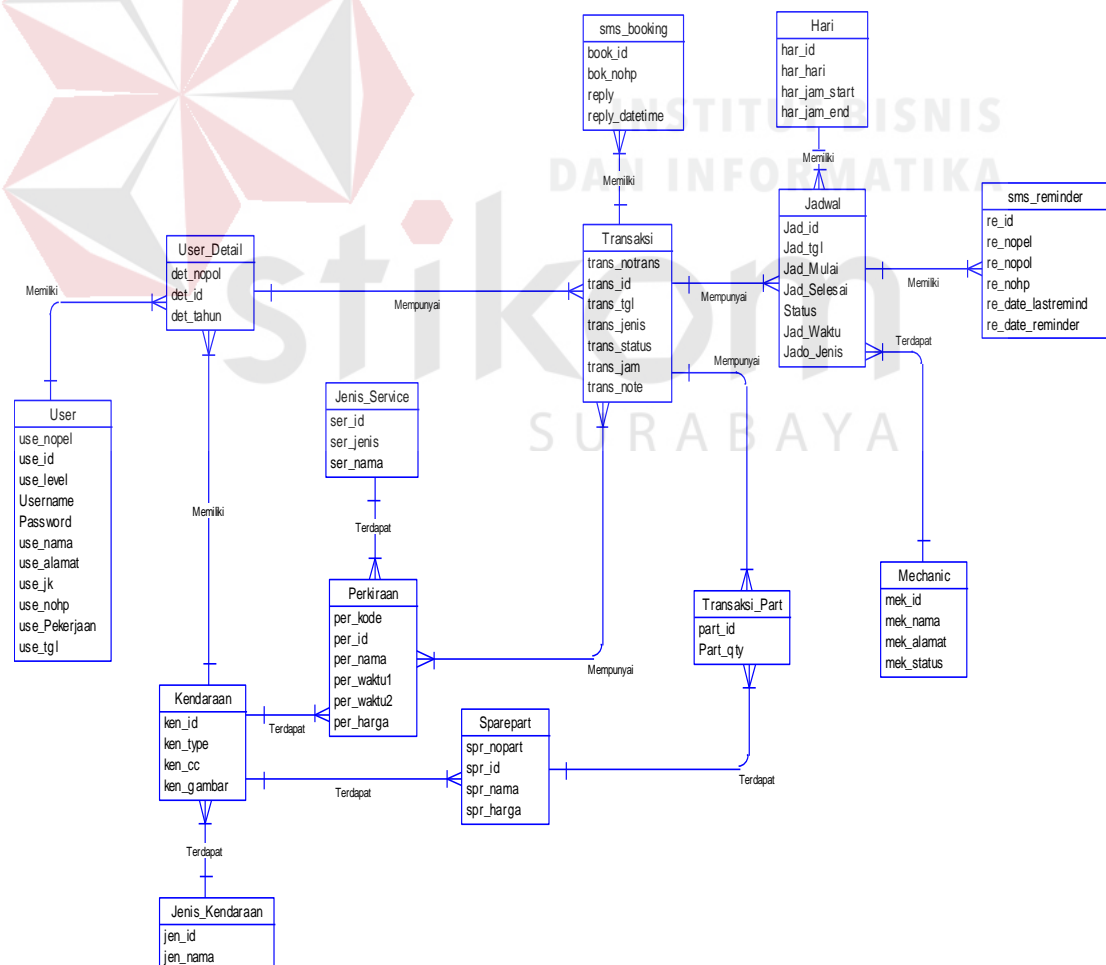
yang telah melakukan proses *booking* dan mengutamakan pelayanan terhadap pelanggan berdasarkan antrian dan prioritas pendeknya suatu proses.

### 3.2.4 Entity Relationship Diagram

*Entity Relationship Diagram* atau biasa disingkat *ERD* adalah suatu desain sistem yang digunakan untuk menggambarkan atau menentukan kebutuhan tabel – tabel *database* dalam sistem. Tabel ini akan digambarkan dalam bentuk *entity* dan memiliki atribut serta saling berhubungan atau relasi satu sama lain.

Penggambaran *ERD* lebih jelasnya adalah sebagai berikut :

#### A. Conceptual Data Model

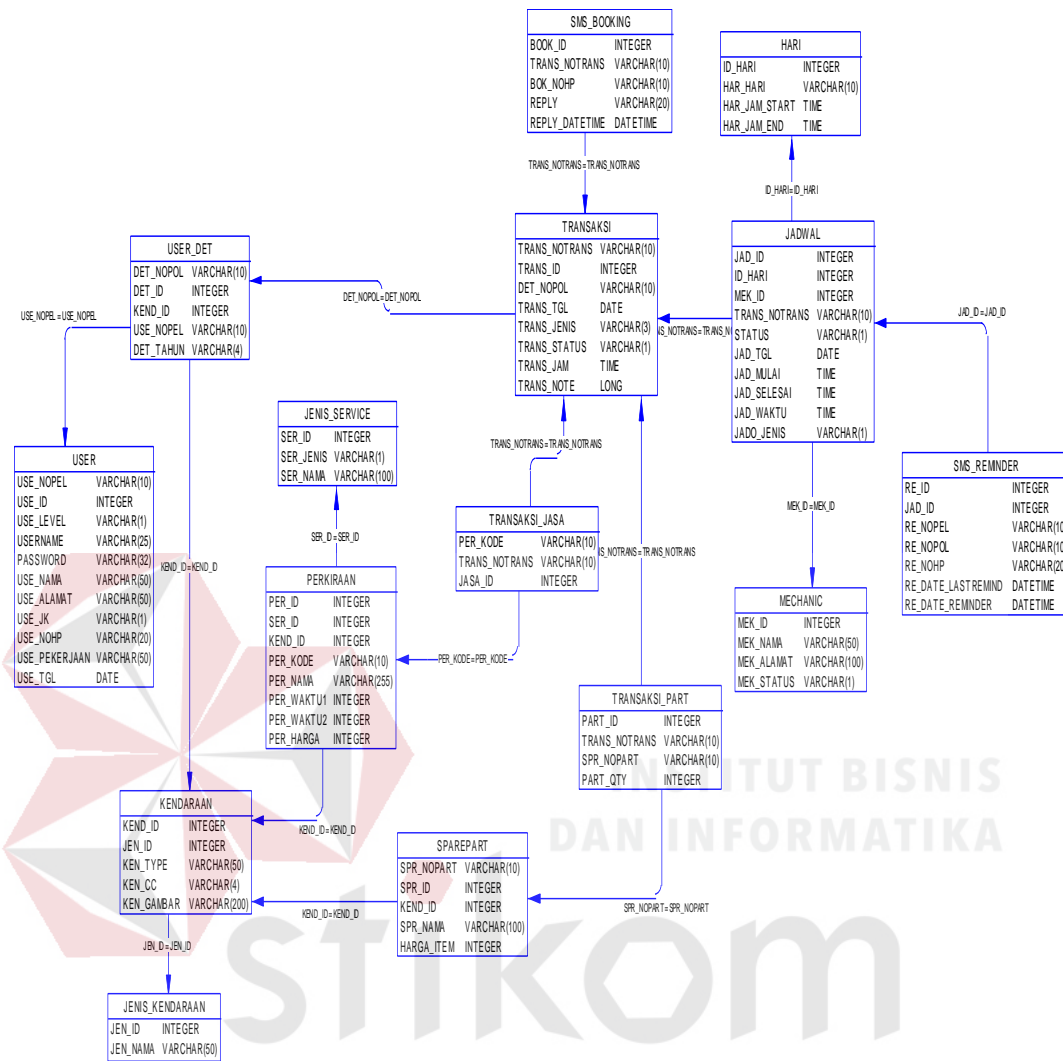


Gambar 3.23 Conceptual Data Model



*Conceptual Data Model* atau biasa disebut dengan *CDM* menggambarkan secara keseluruhan konsep struktur basis data yang dirancang untuk suatu sistem. Pada *CDM* ini sudah dimasukkan beberapa atribut penyusun tiap tabelnya, gunanya menampung data yang terkait didalamnya. Seperti yang terlihat pada gambar diatas, bahwa ada empat belas tabel yang saling berelasi satu sama lain, yaitu tabel jenis kendaraan yang bersifat mandatory bagi tabel-tabel lainnya atau *reference* ke tabel lain, tabel *user* (pelanggan) yang menginputkan data kedalam tabel *user* (pelanggan), tabel jenis *service* memberikan referensi *service* apa yang ingin digunakan, tabel perkiraan waktu sebagai estimasi waktu untuk melakukan *service*, tabel jenis kendaraan adalah jenis kendaraan apa saja yang ada di perusahaan, tabel data kendaraan sebagai tipe kendaraan dari masing jenis kendaraan, tabel detail *user* adalah detail dari data *user* jika *user* (pelanggan) mempunyai kendaraan lebih dari satu, tabel transaksi menginputkan data-data transaksi, tabel part transaksi memberikan data sparepart transaksi jika terdapat pembelian *sparepart*, tabel *sparepart* inputan dari data-data *sparepart*, tabel *mechanic* merupakan inputan dari data-data *mechanic*, tabel hari merupakan hari operasional perusahaan, tabel jadwal merupakan status jadwal *service* apakah sudah terisi atau masih kosong, tabel *sms booking* merupakan konfirmasi kepada pelanggan apakah jadi melakukan perawatan atau tidak dan tabel *sms reminder* sebagai pengingat kepada pelanggan untuk melakukan perawatan kendaraan melalui sms setiap bulannya.

## B. Physical Data Model



Gambar 3.24 Physical Data Model

*Physical Data Model (PDM)* menggambarkan secara detail tentang konsep struktur basis data yang dirancang untuk suatu sistem, yang mana hasil *generate* dari *CDM*. Pada *PDM* ini juga sudah tergambar jelas relasi antar tabelnya, dengan ditunjukkan *primary key* dan *foreign key*nya masing-masing. Sebagai contoh tabel detail *user* yang berelasi dengan tabel *user* dan kendaraan sehingga mendapat *foreign key Det\_ID* dari masing-masing tabel tersebut. Nantinya *PDM* ini akan digenerate untuk menghasilkan *database* dalam *Database Management System (DBMS)*.

### 3.3.6 Struktur Database

Struktur *database* merupakan uraian struktur fisik dari tabel-tabel yang terdapat pada *database*. Fungsinya adalah menyimpan data-data yang saling berhubungan. Adapun struktur *database* tersebut dapat dijelaskan lebih rinci sebagai berikut :

#### A. Tabel User

Nama tabel : tb\_user

Primary key : Use\_nopel

Foreign key : -

Fungsi : Menyimpan data user

**Tabel 3.4 User**

No	Field Name	Data Type	Length	Constraint
1	Use_nopel	Varchar	10	PK
2	Use_id	Integer	4	
3	Use_username	Varchar	25	
4	Use_password	Varchar	32	
5	Use_nopel	Varchar	10	
6	Use_nama	Varchar	50	
7	Use_alamat	Varchar	50	
8	Use_jk	Char	1	
9	Use_nohp	Varchar	20	
10	Use_pekerjaan	Varchar	50	
11	Use_Tgl	Varchar	12	

### B. Tabel Detail User

Nama tabel : tb\_user\_det

Primary key : Det\_id

Foreign key : Use\_nopol ( User ), Ken\_id ( Kendaraan )

Fungsi : Menyimpan data detail user.

**Tabel 3.5** Detail User

No	Field Name	Data Type	Length	Constraint
1	Det_nopol	Varchar	10	PK
2	Det_id	Integer	18	
3	Ken_id	Integer	15	FK
4	Use_nopol	Varchar	10	FK
5	Tahun_pembuatan	Varchar	4	

### C. Tabel Jenis Kendaraan

Nama tabel : tb\_jenis

Primary key : Jen\_id

Foreign key : -

Fungsi : Menyimpan data jenis kendaraan

**Tabel 3.6** Jenis Kendaraan

No	Field Name	Data Type	Length	Constraint
1	Jen_id	Integer	2	PK
2	Jen_nama	Varchar	50	

#### D. Tabel Kendaraan

Nama tabel : tb\_kendaraan

Primary key : Ken\_id

Foreign key : Jen\_id ( Jenis Kendaraan )

Fungsi : Menyimpan data kendaraan

**Tabel 3.7** Kendaraan

No	Field Name	Data Type	Length	Constraint
1	Ken_id	Integer	3	PK
2	Jen_id	Integer	2	FK
3	Ken_type	Varchar	50	
4	Ken_cc	Varchar	4	
5	Ken_gambar	Varchar	200	

#### E. Tabel Jenis Service

Nama tabel : tb\_jenis\_servis

Primary key : Ser\_id

Foreign key : -

Fungsi : Menyimpan data jenis service

**Tabel 3.8** Jenis Service

No	Field Name	Data Type	Length	Constraint
1	Ser_id	Integer	6	PK
2	Ser_nama	Varchar	100	
3	Ser_jenis	Varchar	1	

### F. Tabel *Mechanic*

Nama tabel : *tb\_mekanik*

Primary key : *Mek\_id*

Foreign key : -

Fungsi : Menyimpan data *mechanic*

**Tabel 3.9** *Mechanic*

No	Field Name	Data Type	Length	Constraint
1	<i>Mek_id</i>	Integer	2	PK
2	<i>Mek_nama</i>	Varchar	50	
3	<i>Mek_alamat</i>	Varchar	100	
4	<i>Mek_status</i>	Varchar	1	

### G. Tabel *Sparepart*

Nama tabel : *tb\_sparepart*

Primary key : *Spr\_nopart*

Foreign key : *ken\_id* (kendaraan)

Fungsi : Menyimpan data pembayaran dan pelunasan pinjaman

**Tabel 3.10** *Sparepart*

No	Field Name	Data Type	Length	Constraint
1	<i>Spr_nopart</i>	Varchar	10	PK
2	<i>Ken_id</i>	Integer	3	FK
3	<i>Spr_id</i>	Integer	3	
4	<i>Spr_nama</i>	Varchar	100	
5	<i>Spr_harga</i>	Integer	10	

## H. Tabel Hari

Nama tabel : tb\_hari  
 Primary key : Har\_id  
 Foreign key : -  
 Fungsi : Menyimpan data hari

**Tabel 3.11** Hari

No	Field Name	Data Type	Length	Constraint
1	Har_id	Integer	2	PK
2	Har_hari	Varchar	10	
3	Har_jam_start	Time		
4	Har_jam_end	Time		

## I. Tabel Perkiraan Waktu

Nama tabel : tb\_perkiraan  
 Primary key : Per\_id  
 Foreign key : Ser\_id ( Jenis *Service* ), ken\_id ( Kendaraan )  
 Fungsi : Menyimpan data perkiraan waktu

**Tabel 3.12** Perkiraan

No	Field Name	Data Type	Length	Constraint
1	Per_kode	Varchar	10	PK
2	Ser_id	Integer	6	FK
3	Ken_id	Integer	3	FK
4	Per_id	Integer	6	
5	Per_nama	Varchar	255	
6	Per_waktu1	Integer	2	
7	Per_waktu2	Integer	2	
8	Per_harga	Integer	10	

## J. Tabel Jadwal

Nama tabel : tb\_perkiraan

Primary key : jad\_id

Foreign key : har\_id ( hari ), mek\_id ( *Mechanic* ), trans\_notrans  
(transaksi)

Fungsi : Menyimpan data jadwal

**Tabel 3.13** Jadwal

No	Field Name	Data Type	Length	Constraint
1	jad_id	Integer	10	PK
2	har_id	Integer	10	FK
3	mek_id	Integer	2	FK
4	trans_notrans	Varchar	10	FK
5	jad_tgl	Date		
6	jad_status	Varchar	1	
7	jad_mulai	Time		
8	jad_selesai	Time		
9	jad_waktu	Integer	5	
10	jad_jenis	Varchar	1	

## K. Transaksi

Nama tabel : tb\_transaksi\_h

Primary key : trans\_notrans

Foreign key : det\_nopo ( *Detail User* ), use\_nopol ( *User* )

Fungsi : Menyimpan data transaksi

**Tabel 3.14** Transaksi

No	Field Name	Data Type	Length	Constraint
1	Trans_notrans	Varchar	10	PK
2	Det_nopol	Varchar	10	FK



No	Field Name	Data Type	Length	Constraint
3	Use_nopel	Date	10	FK
4	Trans_id	Integer	10	
5	Trans_tgl	Date		
6	Trans_jenis	Varchar	3	
7	Trans_status	Varchar	25	
8	Trans_note	Text		

#### L. Transaksi Jasa

Nama tabel : tb\_transaksi\_jasa

Primary key : trans\_notrans, per\_id

Foreign key : -

Fungsi : Menyimpan data transaksi jasa

**Tabel 3.15** Transaksi Jasa

No	Field Name	Data Type	Length	Constraint
1	jasa_id	Integer	10	
2	trans_notrans	Integer	10	PK
3	per_id	Integer	6	PK

#### M. Transaksi Part

Nama tabel : tb\_transaksi\_part

Primary key : part\_id

Foreign key : trans\_notrans ( Transaksi ), spr\_nopart ( Sparepart )

Fungsi : Menyimpan data transaksi sparepart

**Tabel 3.16** Transaksi Part

No	Field Name	Data Type	Length	Constraint
1	part_id	Integer	10	PK

No	Field Name	Data Type	Length	Constraint
2	trans_notrans	Varchar	10	FK
3	spr_nopart	Varchar	10	FK
4	part_qty	Integer	10	

#### N. Sms Booking

Nama tabel : tb\_sms\_booking

Primary key : bok\_id

Foreign key : trans\_notrans ( Transaksi )

Fungsi : Menyimpan data booking untuk sms approval

**Tabel 3.17 Sms Booking**

No	Field Name	Data Type	Length	Constraint
1	bok_id	Integer	10	PK
2	trans_notrans	Varchar	10	FK
3	Nohp	Varchar	20	
4	Reply	Char	1	
5	reply_datetime	Datetime		

#### O. Sms Reminder

Nama tabel : tb\_sms\_booking

Primary key : re\_id

Foreign key : jad\_id ( Jadwal )

Fungsi : Menyimpan data booking untuk sms reminder

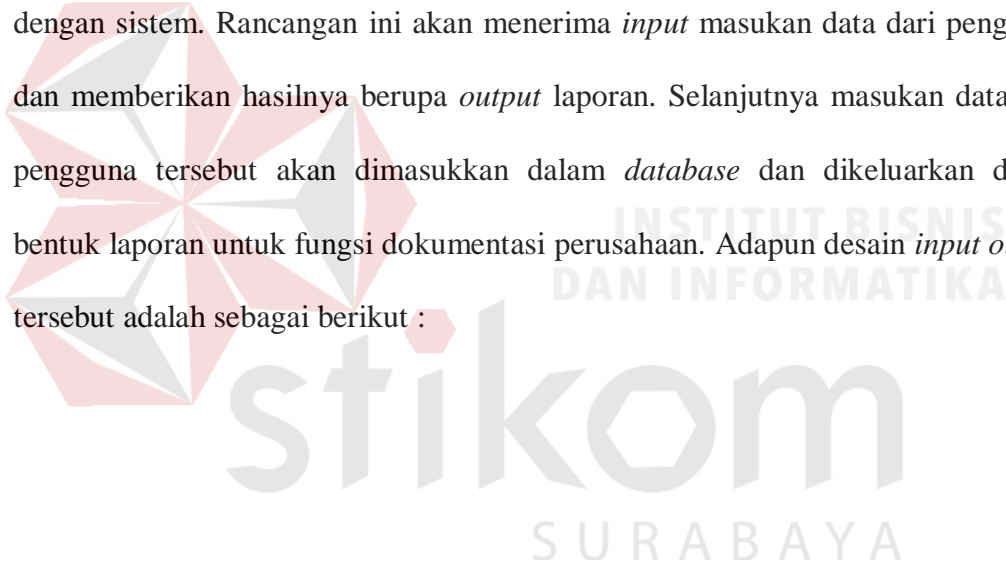
**Tabel 3.18 Sms Reminder**

No	Field Name	Data Type	Length	Constraint
1	re_id	Integer	10	PK
2	jad_id	Integer	10	FK

No	Field Name	Data Type	Length	Constraint
3	Nopel	Varchar	20	
4	Nopol	Varchar	10	
5	Nohp	Varchar	20	
6	date_lastremind	Datetime		
7	date_reminder	Datetime		

### 3.3.7 Desain Input Output

Desain *input output* adalah rancangan form-form yang akan diimplementasikan kedalam sistem dan berfungsi sebagai antar muka pengguna dengan sistem. Rancangan ini akan menerima *input* masukan data dari pengguna dan memberikan hasilnya berupa *output* laporan. Selanjutnya masukan data dari pengguna tersebut akan dimasukkan dalam *database* dan dikeluarkan dalam bentuk laporan untuk fungsi dokumentasi perusahaan. Adapun desain *input output* tersebut adalah sebagai berikut :



## A. Desain *Input*

### A.1 Form *Login*



The image shows a login form for 'Kawasaki'. The form is centered on a white background with a dotted pattern. It features a green header bar with the text 'Kawasaki'. The login form itself has a green title bar 'Log in', a 'Username' field with the text 'text', a 'Password' field with asterisks, and two buttons: 'Lupa Password?' and 'Login'. A green footer bar contains the text '@copyright Abner Mahendra 2014 - All Rights Reserved'.

**Gambar 3.25** Form *Login*

Form *Login* ini digunakan untuk keamanan sistem. Tujuannya adalah supaya sistem digunakan oleh orang yang berhak memakai dan berjalan sesuai hak aksesnya masing-masing. Adapun *field* yang harus diisi dalam form ini adalah *username* dan *password*. Selanjutnya klik tombol *login*, secara otomatis sistem akan bekerja untuk validasi pengguna tersebut dan mencari hak aksesnya. Apabila lupa password klik tombol lupa *password*.

## A.2 Form Menu Utama



**Gambar 3.26** Form Menu Utama

Form Menu Utama ini halaman utama setelah melakukan *login*. Dalam form tersebut terdapat menu utama dimana menu utama tersebut mempunyai sub menu. Seperti contoh menu utama master mempunyai sub menu *user*, jenis kendaraan, kendaraan, jenis *service*, *mechanic*, *sparepart*, hari operasional, perkiraan, *detail user*.

### A.3 Form *User* ( Pelanggan )

**Kawasaki**

Home Master Transaksi Penjadwalan Laporan Logout

Username  Nomor Pelanggan   
 Password  Nama Customer   
 Jenis User Pelanggan  Alamat   
 Jenis Kelamin ..Pilih Jenis Kelamin..   
 No Handphone   
 Pekerjaan

Simpan Batal

Pencarian :  Cari... Refresh

no	Username	Jenis User	No Pelanggan	Nama	Alamat	Kelola
1	xxxxxxx	Pelanggan	xxxxxxx	xxxxxxx	xxxxxxx	<input type="button" value="QI"/> <input type="button" value="edit"/>

@copyright Abner Mahendra 2014 - All Rights Reserved

**Gambar 3.27** Form *User* (Pelanggan)

Form Master *User* (Pelanggan) ini digunakan untuk memasukkan data identitas pelanggan. Dalam data ini terdapat identitas pelanggan antara lain : nomor pelanggan yang *digenerate* otomatis oleh sistem, username, password, jenis user, nama pelanggan, alamat, jenis kelamin, no handphone dan pekerjaan. Selanjutnya data tersebut akan disimpan kedalam database dan *edit* jika diperlukan.

#### A.4 Form Jenis Kendaraan

The screenshot shows a web application interface for 'Kawasaki'. At the top, there is a green header with the name 'Kawasaki'. Below it is a navigation menu with buttons for 'Home', 'Master', 'Transaksi', 'Penjadwalan', 'Laporan', and 'Logout'. The main content area is titled 'Kelola Jenis Kendaraan'. It features a form for entering vehicle type with a text input field and 'Simpan' and 'Batal' buttons. Below the form is a search section with a 'Pencarian : text' input, 'Cari...' and 'Refresh' buttons. A table displays the following data:

no	Jenis Kendaraan	Kelola
1	xxxxxx	<a href="#">QN</a> <a href="#">●</a>

At the bottom of the page, there is a copyright notice: '@copyright Abner Mahendra 2014 - All Rights Reserved'.

**Gambar 3.28** Form Jenis Kendaraan

Form jenis kendaraan ini digunakan untuk memasukkan data jenis kendaraan. Dalam form ini terdapat data jenis kendaraan perusahaan, form yang dibutuhkan antara lain : nomor jenis kendaraan yang *digenerate* otomatis oleh sistem dan jenis kendaraan tersebut. Selanjutnya data tersebut akan disimpan kedalam database dan *edit* jika diperlukan.

## A.5 Form Kendaraan

**Gambar 3.29** Form Kendaraan

Form data kendaraan ini digunakan untuk memasukkan data kendaraan. Dalam form ini terdapat data kendaraan perusahaan berdasarkan jenis kendaraan, form yang dibutuhkan antara lain : nomor kendaraan yang digenerate otomatis oleh sistem, jenis kendaraan yang diambil dari tabel jenis kendaraan, type kendaraan, *cylinder* (cc), gambar sepeda motor tersebut. Selanjutnya data tersebut akan disimpan kedalam database dan *edit* jika diperlukan.



## A.6 Form Jenis Service

The screenshot shows a web application interface for 'Kawasaki'. At the top, there is a navigation menu with links: Home, Master, Transaksi, Penjadwalan, Laporan, and Logout. Below the menu, the page title is 'Jenis Service'. The main form area contains a dropdown menu for 'Jenis Service' with the text '.. Pilih Jenis Service ..', a text input field for 'Nama Servis', and two buttons: 'Simpan' and 'Batal'. Below the form, there is a search section with a text input field containing 'text', and two buttons: 'Cari...' and 'Refresh'. A table displays the following data:

no	Jenis Service	Nama Service	Kelola
1	Ringan - Rutin	xxxxxxx	<a href="#">CN</a> <a href="#">C</a>

At the bottom of the page, there is a copyright notice: '@copyright Abner Mahendra 2014 - All Rights Reserved'.

**Gambar 3.30** Form Jenis Service

Form jenis *service* ini digunakan untuk memasukkan data jenis *service*. Dalam form ini terdapat data jenis *service* perusahaan, form yang dibutuhkan antara lain : nomor jenis *service* yang digenerate otomatis oleh sistem dan jenis *service* (*default* berdasarkan perusahaan) tersebut, nama *service*. Selanjutnya data tersebut akan disimpan kedalam database dan *edit* jika diperlukan.

### A.7 Form *Mechanic*

The screenshot shows a web application interface for 'Kawasaki'. At the top, there is a navigation menu with options: Home, Master, Transaksi, Penjadwalan, Laporan, and Logout. Below the menu, the main heading is 'Kelola Mechanic'. The form contains two input fields: 'Nama Mechanic' and 'Alamat'. To the right of these fields are two buttons: 'Simpan' and 'Batal'. Below the form is a search section with a text input field containing 'text', and two buttons: 'Cari...' and 'Refresh'. Underneath the search section is a table with the following data:

no	Nama Mechanic	Alamat	Status	Kelola
1	xxxxxx	xxxxxxxx	Free	<a href="#">CN</a> <a href="#">+</a>

At the bottom of the page, there is a footer with the text: '@copyright Abner Mahendra 2014 - All Rights Reserved'.

**Gambar 3.31** Form *Mechanic*

Form *mechanic* ini digunakan untuk memasukkan data *mechanic*. Dalam form ini terdapat data *mechanic* perusahaan, form yang dibutuhkan antara lain : nomor *mechanic* yang digenerate otomatis oleh sistem, nama *mechanic*, alamat, status *mechanic* untuk awal berstatus *free*. Selanjutnya data tersebut akan disimpan kedalam database dan *edit* jika diperlukan.

## A.8 Form *Sparepart*

Kawasaki

Home Master Transaksi Penjadwalan Laporan Logout

Kelola Sparepart

Jenis Kendaraan xxxxxx  
Tipe Kendaraan xxxxxx

No Part  
Nama Part  
Harga

Simpan Batal

Pencarian : text Cari... Refresh

no	No Part	Nama Part	Harga	Tipe Kendaraan	Kelola
1	999-9999	xxxxxxx	99.999	xxxxxx	QN

@copyright Abner Mahendra 2014 - All Rights Reserved

**Gambar 3.32** Form *Sparepart*

Form *sparepart* ini digunakan untuk memasukkan data *sparepart*. Dalam form ini terdapat data *sparepart* berdasarkan jenis kendaraan dan tipe kendaraan perusahaan, form yang dibutuhkan antara lain : nomor *sparepart* yang digenerate otomatis oleh sistem, jenis kendaraan yang diambil dari tabel jenis kendaraan, *type* kendaraan yang diambil dari tabel kendaraan, no *part*, nama *part*, harga. Selanjutnya data tersebut akan disimpan kedalam database dan *edit* jika diperlukan.

## A.9 Form Hari

The screenshot shows a web application interface for 'Kawasaki'. At the top, there is a navigation menu with links: Home, Master, Transaksi, Penjadwalan, Laporan, and Logout. Below the menu, the page title is 'Kelola Hari'. The main form area contains a text input field for 'Hari', two sets of dropdown menus for 'Jam Mulai' (0 and 0) and 'Jam Selesai' (0 and 0), and two buttons: 'Simpan' and 'Batal'. Below the form is a search section with a text input field containing 'text', and 'Cari...' and 'Refresh' buttons. A table displays the following data:

no	Hari	Jam Mulai	Jam Selesai	Kelola
1	sssss	08:00:00	17:00:00	<a href="#">QN</a> <a href="#">🔍</a>

At the bottom of the page, there is a footer with the text: '@copyright Abner Mahendra 2014 - All Rights Reserved'.

**Gambar 3.33** Form Hari

Form hari ini digunakan untuk memasukkan data hari. Dalam form ini terdapat data hari *operasional* perusahaan, form yang dibutuhkan antara lain : nomor *hari* yang digenerate otomatis oleh sistem, hari, jam mulai dimana combo box pertama adalah jam sedangkan combo box kedua adalah menit, dan jam selesai dimana combo box pertama adalah jam sedangkan combo box kedua adalah menit. Selanjutnya data tersebut akan disimpan kedalam database dan *edit* jika diperlukan.

## A.10 Form Perkiraan

**Kawasaki**

Home Master Transaksi Penjadwalan Laporan Logout

Tambah Perkiraan Waktu

Kode Pekerjaan

Nama Pekerjaan

Jenis Kendaraan

Tipe Kendaraan

Jenis Perbaikan

Estimasi Perkiraan Waktu  [Jam]  [Menit]

Harga Jasa

Simpan Batal

Pencarian :  Cari... Refresh

no	Kode	Nama Pekerjaan	Jenis	Tipe	Jenis Perbaikan	Estimasi	Biaya Jasa	Kelola
1	xxxxxxx	xxxxxxx	xxxxx	xxxx...	2 jam, 0 menit	999.999		<input type="button" value="QN"/> <input type="button" value="edit"/>

@copyright Abner Mahendra 2014 - All Rights Reserved

**Gambar 3.34** Form Perkiraan

Form perkiraan ini digunakan untuk memasukkan data perkiraan waktu. Dalam form ini terdapat data perkiraan *service* berdasarkan jenis kendaraan, type kendaraan dan jenis perbaikan (jenis *service*) perusahaan. Form yang dibutuhkan antara lain : nomor perkiraan yang digenerate otomatis oleh sistem, jenis kendaraan yang diambil dari tabel jenis kendaraan, *type* kendaraan yang diambil dari tabel kendaraan, jenis perbaikan yang diambil dari tabel jenis *service*, kode pekerjaan, nama pekerjaan, estimasi perkiraan waktu, harga jasa. Selanjutnya data tersebut akan disimpan kedalam database dan *edit* jika diperlukan.

### A.11 Form Detail *User*



**Gambar 3.35** Display Detail *User*

Display detail *user* ini digunakan admin untuk memilih pelanggan agar ditambahkan data kendaraan apa saja yang dimiliki pelanggan, dimana pelanggan bisa saja memiliki satu kendaraan kawasaki atau lebih. Admin dapat menambahkan data kendaraan dan klik di tabel kelola, dan akan muncul gambar selanjutnya pada gambar 3.37.

**Kawasaki**

Home Master Transaksi Penjadwalan Laporan Logout

Kelola Pelanggan

No Pelanggan: sssssss

Nama Pelanggan: ssssssss

Alamat: ssssssss

Jenis Kendaraan: ssssss

Tipe Kendaraan: ssssss

No. Polisi:

Tahun Pembuatan:

Simpan Batal

Pencarian: text Cari... Refresh

no	Nopol	Jenis Kendaraan	Tipe Kendaraan	Tahun
1	ssssss	sssssssss	xxxxxxxx	xxxx

@copyright Abner Mahendra 2014 - All Rights Reserved

**Gambar 3.36** Detail *User*

Form detail *user* ini digunakan untuk memasukkan data detail *user*. Dalam form ini terdapat data detail *user* berdasarkan data user. Form yang dibutuhkan antara lain : nomor detail yang digenerate otomatis oleh sistem, jenis kendaraan yang diambil dari tabel jenis kendaraan, *type* kendaraan yang diambil dari tabel kendaraan, no polisi, tahun pembuatan. Selanjutnya data tersebut akan disimpan kedalam database.

## A.12 Transaksi Administrasi



The screenshot displays the Kawasaki web application interface. At the top, there is a green header with the word "Kawasaki" in white. Below the header is a navigation menu with buttons for "Home", "Master", "Transaksi", "Penjadwalan", "Laporan", and "Logout". The main content area is titled "Tambah Registrasi Service" and contains a form with the label "Masukkan Nopol Kendaraan". The form has a text input field and two buttons: "Batal" and "Proses". At the bottom of the page, there is a green footer with the text "@copyright Abner Mahendra 2014 - All Rights Reserved".

**Gambar 3.37** Form *Input* Transaksi Nopol

Form *input* transaksi nopol ini digunakan kasir untuk memasukkan nomor polisi kendaraan pelanggan sebelum melakukan transaksi. Melalui nomor polisi tersebut sistem akan melakukan check database apakah kendaraan tersebut adalah pelanggan atau tidak. Selanjut jika nomor polisi tersebut adalah pelanggan akan muncul gambar selanjutnya pada gambar 3.39.



**Kawasaki**

Home Master Transaksi Penjadwalan Laporan Logout

Tanggal: 2014-12-03  
 No Registrasi: XXXXXXXXXXXX  
 No Pelanggan: XXXXXXXXXXXX  
 Nama Pelanggan: XXXXXX

Nopol: XXXXXXXX  
 Type Kendaraan: XXXXXXXXXXXX

Pilih Jenis Service: ..Pilih Jenis Service.. [▼] [Tambahkan]

[Batal] [Simpan]

no	Jenis Service	Estimasi	Hapus
1	XXXXXXXXXXXXXXXXXX	1 jam, 30 menit	<input type="checkbox"/>

[Hapus]

@copyright Abner Mahendra 2014 - All Rights Reserved

**Gambar 3.38** Form *Input* Transaksi Administrasi

Form *input* transaksi administrasi ini digunakan untuk memasukkan data transaksi pelanggan yang akan melakukan perawatan kendaraan sepeda motor. Form yang dibutuhkan antara lain : tanggal, no *registrasi*, no pelanggan, nama pelanggan, nopol, tipe kendaraan yang dibaca dari tabel *user*, dan *input*-an berupa *combobox* jenis *service*. Selanjutnya data tersebut akan disimpan kedalam database.

## B. Desain Output

### B.1 Jadwal Service

Tanggal : xx-xx-xx

	08				09				10				11				12			
	00	15	30	45	00	15	30	45	00	15	30	45	00	15	30	45	00	15	30	45
Mekanik 1																				
Mekanik 2																				
Mekanik 3																				
Mekanik 4																				
Mekanik 5																				

Gambar 3.39 Jadwal Service

Jadwal ini digunakan untuk informasi penjadwalan. Selain itu jadwal ini juga bertujuan untuk mengetahui panjangnya jam sebuah proses. Beberapa kolom yang ada pada laporan ini adalah jam, dan mekanik yang ada. Desain jadwal ini dapat dilihat pada gambar diatas.

### B.2 Desain Work Order *Mechanic*

Kawasaki		PT. Surapita Unitrans Surabaya
<b>Work Order</b>		
No Registrasi : XXXXXXXXXXXX		Surabaya , XX-XX-XXXX
Harap dikerjakan pekerjaan tersebut dibawah ini :		
No	Kode Pekerjaan	Jasa Service
1	XXXXXXXXXXXXXX	Servis Berkala / Tune Up - RTN - Pmrksaan tekanan angin
2	XXXXXXXXXXXXXX	Servis Berkala / Tune Up - RTN - Service berkala / Tune up
		Total Pekerjaan : 2
Terbilang # Dua Pekerjaan #		
		Mengetahui
		<b>Chief Mechanic</b>

Gambar 3.40 Desain Work Order

Laporan *work order* ini digunakan untuk dokumentasi pekerjaan yang akan dilakukan *mechanic*. Selain itu laporan ini juga bertujuan untuk mengetahui jasa perawatan sepeda motor apa saja yang akan dilakukan *mechanic*. Beberapa kolom yang ada pada laporan ini adalah nomor, kode pekerjaan, dan jasa *service*. Desain laporan dapat dilihat pada gambar diatas.

### B.3 Desain Nota Tagihan

<b>Kawasaki</b>		PT. Surapita Unitrans Surabaya			
<b>Nota Tagihan</b>		Surabaya , DD-MM-YYYY			
No Registrasi : XXXXXXXXXXXX		Nama Pelanggan : Abner Mahendra			
No Pelanggan : XXXXXXXXXXXX		Tipe Kendaraan : Athlete			
No Polisi : XXXXXXXXXXXX					
Perincian Jasa					
No	Jenis Jasa Service		Biaya Jasa		
1	Servis Berkala / Tune Up - RTN - Service berkala / Tune up		XXXXXXXXXXXXXX		
2	Servis Berkala / Tune Up - RTN - Pmrksaan tekanan angin		XXXXXXXXXXXXXX		
Total Biaya Jasa :			XXXXXXXXXXXXXX		
Perincian Sparepart					
No	Kode Sparepart	Nama Sparepart	Jumlah	Harga	Total
1	XXXXXXXXXX	XXXXXXXXXX	X	XXXXXXXXXX	XXXXXXXXXX
Total Biaya Part :					XXXXXXXXXX
Total Biaya : XXXXXXXXXX					
Terbilang : # XXXXXXXXX XXXXXXXX XXXXX #					
					Hormat Kami
					<b>Kasir</b>

**Gambar 3.41** Desain Nota Tagihan

Nota tagihan ini digunakan untuk dokumentasi pekerjaan yang sudah dikerjakan untuk dilakukan diberikan kepada pelanggan dan dilakukan pembayaran. Selain itu laporan ini juga bertujuan untuk mengetahui mengetahui biaya jasa *service*, biaya *sparepart* seta total biaya keseluruhan. Beberapa kolom yang ada pada perincian jasa adalah nomor, jasa *service*, biaya jasa. Sedangkan untuk perincian sparepart nomor, kode *sparepart*, nama *sparepart*, jumlah

*quantity*, harga, dan total harga. Desain nota tagihan dapat dilihat pada gambar diatas.

#### B.4 Desain Laporan *Service*

<b>Kawasaki</b>		<b>PT. Surapita Unitrans Surabaya</b>							
<b>Laporan Service</b>									
Tanggal : DD-MM-YYYY									
No	Nama Mechanic	Ringan				Berat	1 Unit Service	Biaya	
		KSG	RTN	RGK	LTRK			Service	Sparepart
1	XXXX XXXX	999	999	999	999	999	999	999	999
2	XXXX XXXX	999	999	999	999	999	999	999	999
3	XXXX XXXX	999	999	999	999	999	999	999	999
4	XXXX XXXX	999	999	999	999	999	999	999	999
Jumlah		999	999	999	999	999	999	999	999

**Gambar 3.42** Desain Laporan *Service*

Laporan kerusakan *service* ini digunakan untuk dokumentasi kerusakan *service* apa saja yang dilakukan per periode. Selain itu laporan ini juga bertujuan untuk mengetahui nama pelanggan, tanggal *service*, tipe kendaraan yang melakukan perawatan sepeda motor. Beberapa kolom yang ada pada laporan ini adalah nomor, nama pelanggan, tanggal *service*, nomor polisi, tipe kendaraan, jasa *service*. Desain laporan dapat dilihat pada gambar diatas.

### B.5 Desain Laporan Kerusakan *Service*

<b>Kawasaki</b>		<b>PT. Surapita Unitrans Surabaya</b>				
<b>Laporan Kerusakan Service</b>						
No	Tanggal Service	Nama Pelanggan	No Polisi	Tipe Kendaraan	Jasa Service	Mechanic
1	DD-MM-YYYY	XXXXXX	XXXXXXX	Athlete	Servis Berkala / Tune Up - RTN - Service berkala / Tune up	Hari Panca
					Service berkala / Tune up - RTN - Ongkos ganti oli	
2	DD-MM-YYYY	XXXXXX	XXXXXXX	ER-6n	Servis Berkala / Tune Up - RTN - Service berkala / Tune up	Mulyadi
					Service berkala / Tune up - RTN - Bersihkan saringan udara	
						Total Data : 2

**Gambar 3.43** Desain Laporan Kerusakan *Service*

Laporan kerusakan *service* ini digunakan untuk dokumentasi kerusakan *service* apa saja yang dilakukan per periode. Selain itu laporan ini juga bertujuan untuk mengetahui nama pelanggan, tanggal *service*, tipe kendaraan yang melakukan perawatan sepeda motor. Beberapa kolom yang ada pada laporan ini adalah nomor, nama pelanggan, tanggal *service*, nomor polisi, tipe kendaraan, jasa *service*. Desain laporan dapat dilihat pada gambar diatas.

## B.6 Desain Laporan Pendapatan Jasa Service

<b>Kawasaki</b>		<b>PT. Surapita Unitrans Surabaya</b>		
<b>Laporan Pendapatan Jasa Service</b>				
No	Tanggal Service	Nama Pelanggan	Tipe Kendaraan	Jasa Service & Harga
1	DD-MM-YYYY	XXXXXXXXX	XXXXXX	Service berkala / Tune up 999.999
				Ongkos ganti oli 999.999
				Total Harga : 999.999

**Gambar 3.44** Desain Laporan Pendapatan Jasa Service

Laporan pendapatan jasa *service* ini digunakan untuk dokumentasi pendapatan jasa *service* per periode. Selain itu laporan ini juga bertujuan untuk mengetahui nama pelanggan, tanggal *service*, tipe kendaraan jasa *service*, dan harga jasa. Beberapa kolom yang ada pada laporan ini adalah nomor, nama pelanggan, tanggal *service*, tipe kendaraan, jasa *service* dan harga jasa. Desain laporan dapat dilihat pada gambar diatas.

### B.7 Desain Laporan Pendapatan *Sparepart*

<b>Kawasaki</b>		<b>PT. Surapita Unitrans Surabaya</b>				
<b>Laporan Pendapatan Sparepart</b>						
No	Tanggal Transaksi	Kode Sparepart	Nama Sparepart	Jumlah	Harga	Total
1	DD-MM-YYYY	XXXXXX	XXXX	X	XXXX	XXXX
2	DD-MM-YYYY	XXXXXX	XXXX	X	XXXX	XXXX
3	DD-MM-YYYY	XXXXXX	XXXX	X	XXXX	XXXX
4	DD-MM-YYYY	XXXXXX	XXXX	X	XXXX	XXXX
Total Harga :						XXXX

**Gambar 3.45** Desain Laporan Pendapatan *Sparepart*

Laporan pendapatan jasa *service* ini digunakan untuk dokumentasi pendapatan jasa *service* per periode. Selain itu laporan ini juga bertujuan untuk mengetahui nama pelanggan, tanggal *service*, tipe kendaraan jasa *service*, dan harga jasa. Beberapa kolom yang ada pada laporan ini adalah nomor, nama pelanggan, tanggal *service*, tipe kendaraan, jasa *service* dan harga jasa. Desain laporan dapat dilihat pada gambar diatas.

## B.8 Desain Laporan Pekerjaan *Mechanic*

<b>Kawasaki</b>		PT. Surapita Unitrans Surabaya			
<b>Laporan Pekerjaan Mechanic</b>					
Mechanic : XXXXX			Tanggal : XX-XX-XXXX		
No	Type Kendaraan	Nopol	Tanggal	Nama Pelanggan	Jasa Service & Harga
1	XXXX	XXXXXX	DD-MM-YYYY	XXXXXX	XXXX 999.999
2	XXXX	XXXXXX	DD-MM-YYYY	XXXXXX	XXXX 999.999
Total Harga :					999.999

**Gambar 3.46** Desain Laporan Pekerjaan *Mechanic*

Laporan pekerjaan *mechanic* ini digunakan untuk dokumentasi setiap pekerjaan *mechanic* yang telah dikerjakan. Selain itu laporan ini juga bertujuan untuk mengetahui setiap pekerjaan yang telah dilakukan *mechanic*. Desain laporan dapat dilihat pada gambar diatas.



## BAB IV

### IMPLEMENTASI DAN EVALUASI

#### 4.1 Kebutuhan Sistem

Tahap implementasi sistem merupakan tahap pembuatan perangkat lunak yang disesuaikan dengan rancangan atau desain sistem yang telah dibuat. Adapun kebutuhan sistem terhadap perangkat keras dan perangkat lunak supaya bisa berjalan dengan baik adalah sebagai berikut :

##### 4.1.1 *Software dan Hardware untuk Server*

*Software dan hardware untuk server* saat melakukan testing pada aplikasi ini adalah:

- a. Sistem Operasi Windows 7.
- b. *Apache + MySQL* ( Paket XAMPP ).
- c. Memory RAM 2 GB.
- d. Satu *Hardisk* kapasitas *memory* 229 Gb.
- e. Satu *mouse*, dan *keyboard* yang kompatibel.
- f. *Software sms now for windows*.
- g. Modem GSM.

##### 4.1.2 *Software dan Hardware Client*

*Software dan hardware untuk client* saat melakukan testing pada aplikasi ini adalah:

- a. Sistem Operasi *Windows 7*.
- b. *Processor Intel Core 2 Duo*.
- c. Memory RAM 2 GB.
- d. Satu *mouse*, dan *keyboard* yang kompatibel.
- e. *Wi-fi* untuk koneksi *Client* ke server
- f. Salah satu *web-browser* seperti *internet explorer*, *mozilla* dan lain-lain.

## 4.2 Implementasi Sistem

Maka proses selanjutnya adalah implementasi atau penerapan sistem. Implementasi sistem ini merupakan proses Aplikasi Pelayanan Administrasi Perawatan Sepeda Motor di PT. Surapita Unitrans Surabaya. Agar proses implementasi aplikasi menjadi lebih baik, mudah dan terorganisir, maka proses implementasi dikelompokkan dengan beberapa langkah-langkah yang ada.

### 1. Petugas Admin

Petugas admin merupakan *super-user* dimana admin dapat menambahkan data master, transaksi, *view* jadwal *service* dan dapat melakukan *approval* sebelum dilaporkan kepada kepala divisi

### 2. Petugas Kasir

Petugas kasir dapat menambahkan data transaksi pelanggan dan hanya dapat melihat data master didalam aplikasi ini

### 3. Kepala Divisi

Kepala divisi hanya dapat mengakses setiap laporan, baik laporan master, laporan *service*, laporan kerusakan *service*, laporan pendapatan jasa, laporan

pendapatan *sparepart*, dan laporan pekerjaan *mechanic*, untuk keperluan perusahaan

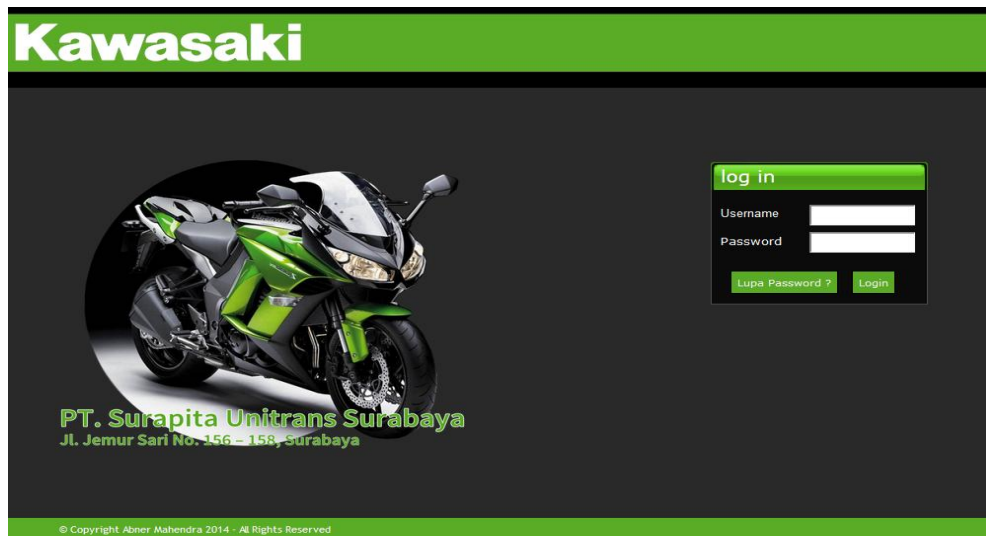
#### 4. Pelanggan

Pelanggan dapat menambahkan *booking service* motor sesuai dengan jam dan hari yang dipilih oleh pelanggan, pelanggan juga dapat melihat *view* jadwal dan *history service* untuk mengetahui *service* apa saja yang telah dilakukan pelanggan

Dari implementasi ini maka Form awal yang akan tampil dalam sistem ketika dijalankan adalah form *login*, sebagai form keamanan bagi pengguna yang berhak untuk mengaksesnya.

##### 4.2.1 Form *Login*

Form *login* digunakan pengguna untuk masuk kedalam sistem dan berguna sebagai proses keamanan sistem bagi pengguna yang berhak mengakses. Dalam form ini pengguna harus memasukkan *username* dan *password* pada *field* yang telah disediakan, kemudian klik *Login*. Sistem akan mengecek *account* tersebut kedalam *database*, apabila *account* telah terdaftar, maka sistem akan melanjutkan proses menuju menu utama, namun apabila belum terdaftar maka sistem akan menolaknya.



Gambar 4.1 Form *Login*

#### 4.2.2 Form Menu Utama

Setelah sukses melakukan *login* akan keluar Form menu utama, dimana form ini berisi menu-menu yang dapat digunakan oleh masing-masing pengguna sesuai hak aksesnya masing-masing. Keterangan pengguna yang sedang mengakses sistem ini tertera pada bagian pojok kiri atas sistem. Adapun hak akses yang diberikan kepada pengguna sesuai dengan jabatannya adalah admin, kasir, kepala divisi, dan *user* (pelanggan). Form menu utama dapat dilihat pada gambar 4.2.



**Gambar 4.2** Form Menu Utama

## **A. Form Master**

Form master digunakan admin atau kasir untuk melakukan input data master yang ada dalam perusahaan. Form master terdiri dari beberapa sub-sistem antara lain: pendaftaran *user*, jenis kendaraan, kendaraan, jenis *service*, *mechanic*, *sparepart*, hari operasional, perkiraan waktu, detail *user*.

### **A.1. Form Master Pendaftaran *User* (pelanggan)**

Form *input* master *user* (pelanggan) digunakan oleh admin atau kasir untuk pendaftaran data identitas *user*. Form ini terdapat pada sub-sistem master yang ada pada menu utama. Beberapa *field* yang ada didalamnya adalah *username*, *password*, nomor pelanggan, nama, alamat, jenis kelamin, no handphone, pekerjaan. Data hasil *inputan user* ini selanjutnya akan dimasukkan kedalam *database* dan sebagai data master. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai yang diinginkan *user*. Form master pendaftaran *user* dapat dilihat pada gambar 4.3

**Gambar 4.3** Form *Input* Master *User* (Pelanggan)

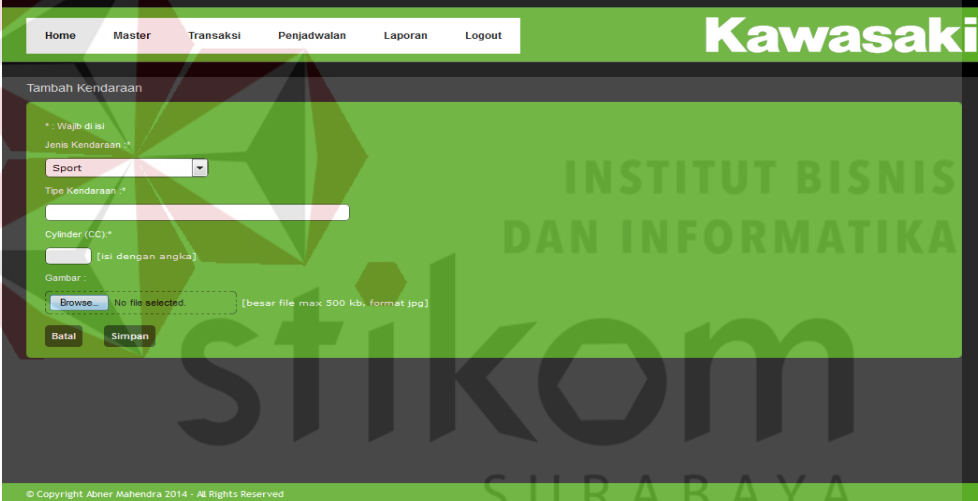
## A.2. Form Jenis Kendaraan

Form *input* jenis kendaraan digunakan oleh admin untuk memasukkan data jenis kendaraan. Form ini terdapat pada sub-sistem master yang ada pada menu utama. *Field* yang ada didalamnya adalah jenis kendaraan. Data hasil *inputan* jenis kendaraan ini selanjutnya akan dimasukkan kedalam *database* dan sebagai data master. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai kebutuhan perusahaan. For jenis kendaraan dapat dilihat pada gambar 4.4

**Gambar 4.4** Form *Input* Jenis Kendaraan

### A.3. Form Kendaraan

Form *input* kendaraan digunakan oleh admin untuk memasukkan data kendaraan berdasarkan jenis kendaraan, sedangkan kasir hanya dapat melihat data kendaraan saja. Form ini terdapat pada sub-sistem master yang ada pada menu utama sebelumnya. Beberapa *field* yang ada didalamnya adalah jenis kendaraan, kendaraan, CC, gambar kendaraan. Data hasil *inputan* kendaraan ini selanjutnya akan dimasukkan kedalam *database* dan sebagai data master. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai kebutuhan perusahaan. Form kendaraan dapat dilihat pada gambar 4.5



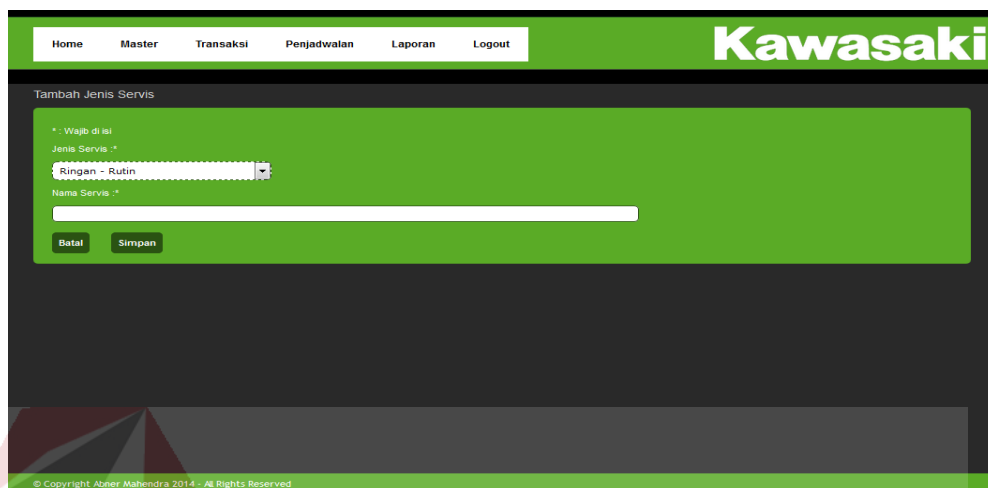
The image shows a web application interface for adding a vehicle. At the top, there is a navigation menu with 'Home', 'Master', 'Transaksi', 'Penjadwalan', 'Laporan', and 'Logout'. The main header is green with the 'Kawasaki' logo. Below the header, the form is titled 'Tambah Kendaraan'. It contains several input fields: 'Jenis Kendaraan' (a dropdown menu with 'Sport' selected), 'Tipe Kendaraan' (a text input field), 'Cylinder (CC)' (a text input field with a note '[isi dengan angka]'), and 'Gambar' (a file upload field with a 'Browse...' button and a note '[besar file max 500 kb, format jpg]'). There are 'Batal' and 'Simpan' buttons at the bottom of the form. A large watermark 'stikom SURABAYA' is visible in the background of the form area. At the bottom left, there is a small copyright notice: '© Copyright Abner Mahendra 2014 - All Rights Reserved'.

**Gambar 4.5** Form *Input* Kendaraan

### A.4. Form Jenis Service

Form *input* jenis *service* digunakan oleh admin untuk memasukkan data jenis *service* yang ada pada perusahaan tersebut, sedangkan kasir hanya dapat melihat data jenis *service* saja. Form ini terdapat pada sub-sistem master yang ada pada menu utama sebelumnya. Beberapa *field* yang ada didalamnya adalah jenis *service* dan nama *service*. Data hasil *inputan* jenis *service* ini selanjutnya akan

dimasukkan kedalam *database* dan sebagai data master. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai kebutuhan perusahaan. Form jenis *service* dapat dilihat pada gambar 4.6



**Gambar 4.6** Form *Input* Jenis *Service*

#### **A.5. Form *Mechanic***

Form *input mechanic* digunakan oleh admin untuk memasukkan data identitas *mechanic*, sedangkan kasir hanya dapat melihat data *mechanic* saja. Form ini terdapat pada sub-sistem master yang ada pada menu utama sebelumnya. Beberapa *field* yang ada didalamnya adalah nama *mechanic* dan alamat. Data hasil *inputan mechanic* ini selanjutnya akan dimasukkan kedalam *database* dan sebagai data master. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai kebutuhan perusahaan. Form *mechanic* dapat dilihat pada gambar 4.7



**Gambar 4.7** Form *Input Mechanic*

### A.6. Form *Sparepart*

Form *input sparepart* digunakan oleh admin untuk memasukkan data *sparepart* berdasarkan jenis kendaraan dan tipe kendaraan, sedangkan kasir hanya dapat melihat data *sparepart* saja. Form ini terdapat pada sub-sistem master yang ada pada menu utama sebelumnya. Beberapa *field* yang ada didalamnya adalah jenis kendaraan, kendaraan, no *part*, harga. Data hasil *inputan sparepart* ini selanjutnya akan dimasukkan kedalam *database* dan sebagai data master. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai kebutuhan perusahaan. Form *sparepart* dapat dilihat pada gambar 4.8

**Gambar 4.8** Form *Input Sparepart*

### A.7. Form Hari Operasional

Display hari operasional ini digunakan admin atau kasir untuk mengetahui hari operasional perusahaan berdasarkan jam mulai dan jam selesai perusahaan. Form ini terdapat pada sub-sistem master yang ada pada menu utama sebelumnya. Admin dapat menambahkan data *sparepart* dengan klik Add Data yang ada dipojok kiri bawah. Form ini dapat dilihat pada gambar 4.9

No.	Hari	Jam Mulai	Jam Selesai	Kelola
1.	Sabtu	08:00:00	14:00:00	
2.	Jumat	08:00:00	17:00:00	
3.	Kamis	08:00:00	17:00:00	
4.	Rabu	08:00:00	17:00:00	
5.	Selasa	08:00:00	17:00:00	
6.	Senin	08:00:00	17:00:00	

Total Record : 6

Gambar 4.9 Display Hari Operasional

### A.8. Form Perkiraan

Form *input* perkiraan digunakan oleh admin untuk memasukkan data perkiraan waktu *service* berdasarkan jenis kendaraan dan tipe kendaraan, sedangkan kasir hanya dapat melihat data perkiraan saja. Form ini terdapat pada sub-sistem master yang ada pada menu utama sebelumnya. Beberapa *field* yang ada didalamnya adalah kode pekerjaan, nama pekerjaan, jenis kendaraan, kendaraan, jenis perbaikan, *estimasi* waktu perbaikan, harga jasa. Data hasil *inputan* perkiraan ini selanjutnya akan dimasukkan kedalam *database* dan sebagai data

master. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai kebutuhan perusahaan Form perkiraan dapat dilihat pada gambar 4.10

The screenshot shows a web form titled 'Tambah Perkiraan Waktu' on the Kawasaki website. The form is set against a green background and contains the following fields and options:

- Kode Pekerjaan \***: JS 20-010
- Nama Pekerjaan \***: Penyetelan rantai
- Jenis Kendaraan \***: Sport
- Tipe Kendaraan \***: ER-6n
- Jenis Perbaikan \***: Service berkala / Tune up - RTN
- Estimasi Waktu Perbaikan**: 0 [jam] 15 [menit]
- Harga Jasa \***: 9000

Buttons for 'Batal' and 'Simpan' are located at the bottom of the form. A copyright notice for Abner Mahendra 2014 is visible at the bottom left.

Gambar 4.10 Form *Input* Perkiraan Waktu *Service*

### A.9. Form Detail *User*

Display detail *user* ini digunakan admin atau kasir untuk mengetahui data identitas detail *user*. Form ini terdapat pada sub-sistem master yang ada pada menu utama. Admin dapat menambahkan data detail dengan memilih salah satu *user* dan klik di kelola kendaraan yang ada di kanan list *user* dan akan muncul form selanjutnya. Display detail *user* pada gambar 4.11.

The screenshot shows a table titled 'Kelola Kepemilikan Kendaraan' on the Kawasaki website. The table has the following columns: No., User Name, Jenis User, No. Pelanggan, Nama, Alamat, and Kelola Kendaraan. The data is as follows:

No.	User Name	Jenis User	No. Pelanggan	Nama	Alamat	Kelola Kendaraan
1.	jova	pelanggan	KS00000005	Josatat Fajar S	Kenjeran 209 Surabaya	[U]
2.	samuel	pelanggan	KS00000004	Samuel Shiba	Runglut 61/100	[U]
3.	matheus	pelanggan	KS00000003	Matheus Gilang P	Jl. Karang Empat Besar no 200	[U]
4.	jono	pelanggan	KS00000002	Herjono Arman	Jl Ketintang Baru Regency Blok A2 No. 2 Surabaya	[U]
5.	abner	pelanggan	KS00000001	Abner Mahendra	Jl Raya Kenjeran 14 Surabaya	[U]

At the bottom of the table, there is a 'Close' button and a 'Total Rekord : 7' indicator. A copyright notice for Abner Mahendra 2014 is visible at the bottom left.

Gambar 4.11 Display Detail *User*

Form detail *user* digunakan oleh admin atau kasir untuk mengetahui data detail *user*. Beberapa *field* yang ada didalamnya nomor pelanggan, nama pelanggan dan alamat pelanggan. Dari form tersebut admin dapat menambahkan data kendaraan pelanggan dengan klik tambahkan kendaraan dan akan muncul form selanjutnya. Detail *user* pada gambar 4.12.

The screenshot displays a web interface for a Kawasaki system. At the top, there is a green navigation bar with the 'Kawasaki' logo on the right and a menu with 'Home', 'Master', 'Transaksi', 'Penjadwalan', 'Laporan', and 'Logout'. Below the navigation bar, the main content area is titled 'Tambah Kepemilikan Kendaraan'. This section contains a form with three input fields: 'Nomor Pelanggan' with the value 'K500000004', 'Nama Pelanggan' with the value 'Samuel Shiba', and 'Alamat' with the value 'Rungkut 61/100'. Below the form are two buttons: 'Back' and 'Tambah Kendaraan'. Underneath the form is a table titled 'Data Kendaraan' with the following columns: 'No.', 'Nopol', 'Jenis', 'Tipe', and 'Tahun'. The page footer includes a copyright notice: '© Copyright Abner Mahendra 2014 - All Rights Reserved'.

**Gambar 4.12** Detail *User*

Form *input* detail *user* digunakan oleh admin atau kasir untuk memasukkan data detail *user* termasuk lanjutan dari form detail *user*. Beberapa *field* yang ada didalamnya jenis kendaraan, tipe kendaraan, no polisi, tahun pembuatan. Data hasil *inputan* detail *user* penambahan kendaraan ini selanjutnya akan dimasukkan kedalam *database* dan sebagai data master detail *user*. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai kebutuhan perusahaan.

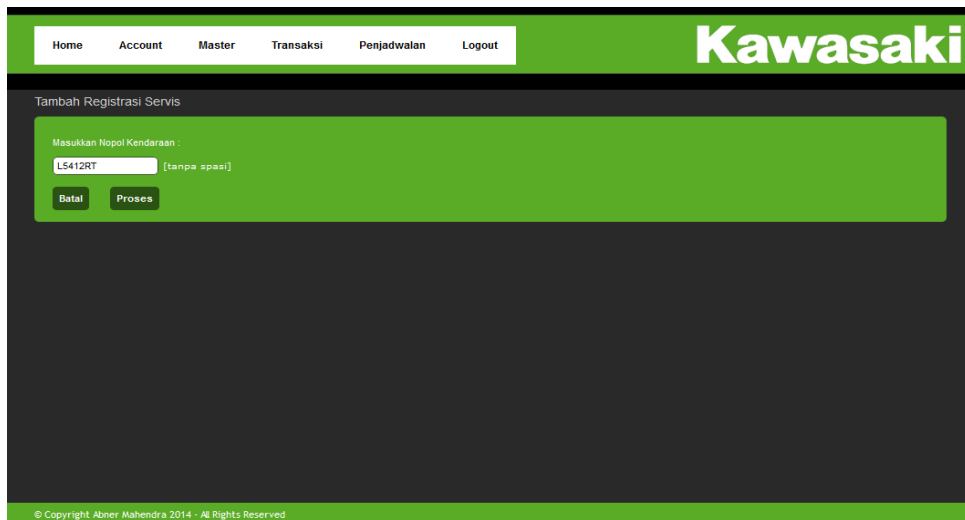
**Gambar 4.13** Form *Input* Detail *User* Penambahan Kendaraan

## B. Form Transaksi

Form transaksi digunakan admin atau kasir untuk melakukan *input* data transaksi perawatan sepeda motor pelanggan. Form transaksi terdiri dari beberapa sub-sistem antara lain: *regular* transaksi, *re-schedule booking*, pemakaian *sparepart*, dan pembayaran.

### B.1. Form Transaksi Administrasi

Untuk menjalankan form transaksi administrasi, pengguna harus memilih menu *regular* transaksi dan melakukan *input* nopol terlebih dahulu sebelum melakukan *service*. Form ini terdapat pada sub-sistem transaksi yang ada pada menu utama. Data hasil *inputan* akan *dicheck* kedalam database apakah dia sudah terdaftar sebagai pelanggan atau tidak. Form untuk input nopol dapat dilihat pada gambar 4.14.



**Gambar 4.14** Form *Input* Nopol

Setelah data *valid* maka pengguna aplikasi ini dapat memasukkan data transaksi *service*. Untuk pelanggan yang melakukan *booking* dapat memilih hari dan jam sesuai keinginan, sedangkan untuk admin hanya bisa melakukan *service* pada hari itu juga. Beberapa *field* yang ada didalamnya adalah tanggal, nomor registrasi, nomor pelanggan, nopol, tipe kendaraan. Data hasil *inputan* transaksi administrasi ini selanjutnya akan dimasukkan kedalam *database* dan disimpan sebagai data transaksi. sistem ini juga menyediakan fungsi *edit* untuk mengubahnya sesuai kebutuhan perusahaan. Form transaksi dapat dilihat pada gambar 4.15

Tambah Registrasi Servis

Tanggal : 2014-12-22  
 Nomor Registrasi : REG0000010  
 Nomor Pelanggan : KS00000004  
 Nama Pelanggan : Jeko Suryanto

Nopol : L5412RT  
 Type Kendaraan : ER-6n

Batal Simpan

Tambah Jenis Servis Hapus Jenis Servis

No.	Jenis Servis	Estimasi Pengerjaan	Biaya
1.	Service berkala / Tune up - RTN - Penyetelan kopling	0 jam, 30 menit	18,000
2.	Service berkala / Tune up - RTN - Service berkala / Tune up	1 jam, 30 menit	300,000

Total Jenis Servis : 2

**Gambar 4.15** Form *Input* Transaksi Administrasi

## B.2. Nota Tagihan

Nota tagihan terdapat pada sub-sistem menu transaksi yang ada pada form menu utama. Nota tagihan ini digunakan untuk dokumentasi pekerjaan yang sudah dikerjakan untuk diberikan kepada pelanggan dan dilakukan pembayaran. Selain itu laporan ini juga bertujuan untuk mengetahui mengetahui biaya jasa *service*, biaya *sparepart* seta total biaya keseluruhan. Beberapa kolom yang ada pada perincian jasa adalah nomor, jenis jasa *service*, biaya jasa. Sedangkan untuk perincian sparepart nomor, kode *sparepart*, nama *sparepart*, jumlah *quantity*, harga, dan total harga. Nota tagihan dapat dilihat pada gambar 4.16

**Kawasaki****INVOICE**

No. Registrasi : **REG0000006** Surabaya, 31-03-2015  
 No. Pelanggan : KS00000002 Nama Pelanggan : Herjono Arman  
 No. Pol : L1234HH Type Kendaraan : ER-6n

Perincian Jasa :		
No.	Jenis Jasa Servis	Biaya Jasa
1.	Servis Berkala / Tune Up - KSG - Sinkronisasi carburator / injeksi	81,000
2.	Servis Berkala / Tune Up - KSG - Service berkala / Tune up	300,000
Total Biaya Jasa :		381,000

Perincian sparepart :					
No.	Kode Sparepart	Nama Sparepart	Jumlah	Harga	Total
1.	13159-1056	CONNECTOR	1	11,000	11,000
Total Biaya Part :					11,000

Total Biaya : 392,000

Terbilang : # Tiga Ratus Sembilan Puluh Dua Ribu Rupiah #

Mengetahui,

Hormat Kami,

**Admin**

**Kasir**

**Note :**

- Invoice ini sebagai bukti sah atas pembayaran.

**Gambar 4.16** Nota Tagihan

### C. Form Penjadwalan

Firm penjadwalan digunakan admin, kasir atau pelanggan untuk melihat jadwal service yang terjadi pada perusahaan. Form penjadwalan terdiri dari beberapa sub-sistem antara lain: penjadwalan hari ini dan penjadwalan booking

#### C.1. Halaman Jadwal Service

Halaman jadwal *service* digunakan oleh pengguna aplikasi ini untuk melihat jadwal service yang dilakukan pada jam dan mekanik siapa saja yang akan menangani *service* tersebut. Hasil tersebut dapat dilihat pada gambar 4.23



12 Mar 2015

	08		09		10		11		12		13		14		15		16																									
	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50
Mekanik 1	REG0000024		REG0000030		REG0000034		REG0000039		REG0000043																																	
Mekanik 2	REG0000025		REG0000033		REG0000038		REG0000044		REG0000046																																	
Mekanik 3	REG0000026		REG0000031		REG0000037		REG0000041																																			
Mekanik 4	REG0000027		REG0000032		REG0000036		REG0000042																																			
Mekanik 5	REG0000028		REG0000029		REG0000035		REG0000040		REG0000045																																	

**Gambar 4.17** Halaman Jadwal *Service*

## D. Laporan

Laporan digunakan *kadiv* atau admin sebagai dokumentasi *service* secara berkala untuk keperluan perusahaan. Laporan terdiri dari beberapa sub-sistem antara lain: laporan *service*, laporan kerusakan *service*, laporan pendapatan jasa, laporan pendapatan *sparepart*, dan laporan pekerjaan *mechanic*

### D.1. Laporan *Service*

Laporan *service* ini digunakan untuk dokumentasi *service* secara keseluruhan pekerjaan *mechanic* berdasarkan pilihan *service* yang telah ditentukan perusahaan, serta adanya informasi biaya *service* dan *sparepart*. Desain laporan dapat dilihat pada gambar 4.18.

**Kawasaki** PT.Surapita Unitrans Surabaya

**Laporan Service**

Tanggal : 31-03-2015

No.	Nama Mekanik	Ringan				Berat	Biaya	
		KSG	RTN	RGK	LTRK		Service	Sparepart
1	Hari Panca	0	5	0	0	0	239,000	58,000
2	Susilo Dwi Nugroho	0	5	0	0	0	621,000	161,000
3	Muliyadi	2	3	0	0	0	410,000	0
4	Indra Permana	0	3	0	0	0	380,000	0
5	Cahyadi Anwar	2	2	0	0	0	425,000	69,000
Jumlah :		4	18	0	0	0	2,075,000	288,000

Cetak Printer Close

Gambar 4.18 Laporan Service

## D.2. Laporan Kerusakan Service

Laporan kerusakan *service* ini digunakan untuk dokumentasi kerusakan *service* apa saja yang dilakukan per periode. Selain itu laporan ini juga bertujuan untuk mengetahui nama pelanggan, tanggal *service*, tipe kendaraan yang melakukan perawatan sepeda motor. Beberapa kolom yang ada pada laporan ini adalah nomor, nama pelanggan, tanggal *service*, nomor polisi, tipe kendaraan, jasa *service*. Desain laporan dapat dilihat pada gambar 4.26.

**Kawasaki** PT.Surapita Unitrans Surabaya

**Laporan Kerusakan Service**

Tanggal : 31-03-2015 s/d 31-03-2015

No.	Nama Pelanggan	Tanggal Servis	Nomor Polisi	Tipe Kendaraan	Jasa Service	Mekanik
1	Abner Mahendra	2015-03-31	W5442GL	Athlete	- Bersihkan karburator - Penyetelan kopling - Service berkala / Tune up	Hari Panca
2	Herjono Arman	2015-03-31	W5623FF	ER-6n	- Ongkos ganti oli - Sinkronisasi carburator / injeksi - Service berkala / Tune up	Susilo Dwi Nugroho
3	Herjono Arman	2015-03-31	W5632JL	Edge R	- Penyetelan rantai - Bersihkan karburator - Service berkala / Tune up	Muliyadi
4	Abner Mahendra	2015-03-31	L5211EW	Pulsar 200NS	- Ongkos ganti oli - Service berkala / Tune up	Indra Permana
5	Herjono Arman	2015-03-31	W2536KL	Athlete	- Penyetelan rantai - Service berkala / Tune up	Cahyadi Anwar
6	Herjono Arman	2015-03-31	L1234HH	ER-6n	- Sinkronisasi carburator / injeksi - Service berkala / Tune up	Cahyadi Anwar
7	Doni Kiswanto	2015-03-31	W4521GH	Ninja RR Special Edition	- Ongkos ganti oli - Service berkala / Tune up	Hari Panca
8	Doni Kiswanto	2015-03-31	W5231TR	Ninja ZX-6R	- Ongkos ganti oli - Service berkala / Tune up	Muliyadi
9	Doni Kiswanto	2015-03-31	W1AR	Z800	- Service berkala / Tune up	Indra Permana
10	Joko Suryanto	2015-03-31	L4521GT	Ninja 250 New SE + ABS	- Sinkronisasi carburator / injeksi - Service berkala / Tune up	Susilo Dwi Nugroho

Total Data : 10

**Gambar 4.19** Laporan Kerusakan Service

### D.3. Laporan Pendapatan Jasa Service

Laporan pendapatan jasa *service* ini digunakan untuk dokumentasi pendapatan jasa *service* per periode. Selain itu laporan ini juga bertujuan untuk mengetahui nama pelanggan, tanggal *service*, tipe kendaraan jasa *service*, dan harga jasa. Beberapa kolom yang ada pada laporan ini adalah nomor, nama pelanggan, tanggal *service*, tipe kendaraan, jasa *service* dan harga jasa. Desain laporan dapat dilihat pada gambar 4.20.

**Kawasaki** PT.Surapita Unitrans Surabaya

**Laporan Pendapatan Jasa Service**

Tanggal : 31-03-2015 s/d 31-03-2015

No.	Nama Pelanggan	Tanggal Servis	Tipe Kendaraan	Jasa Service & Harga	
1	Abner Mahendra	2015-03-31	Athlete	<u>Jasa Service</u>	<u>Harga Service</u>
				Bersihkan karburator	30,000
				Penyetelan kopling	6,000
				Service berkala / Tune up	38,000
2	Herjono Arman	2015-03-31	ER-6n	<u>Jasa Service</u>	<u>Harga Service</u>
				Ongkos ganti oli	36,000
				Sinkronisasi carburator / injeksi	81,000
				Service berkala / Tune up	300,000
3	Herjono Arman	2015-03-31	Edge R	<u>Jasa Service</u>	<u>Harga Service</u>
				Penyetelan rantai	6,000
				Bersihkan karburator	30,000
				Service berkala / Tune up	38,000
4	Abner Mahendra	2015-03-31	Pulsar 200NS	<u>Jasa Service</u>	<u>Harga Service</u>
				Ongkos ganti oli	10,000
				Service berkala / Tune up	70,000
5	Herjono Arman	2015-03-31	Athlete	<u>Jasa Service</u>	<u>Harga Service</u>
				Penyetelan rantai	6,000
				Service berkala / Tune up	38,000
				Total Harga :	689,000

**Gambar 4.20** Laporan Pendapatan Jasa Service

#### D.4. Laporan Pendapatan Sparepart

Laporan pendapatan *sparepart* ini digunakan untuk dokumentasi pendapatan *sparepart* per periode. Selain itu laporan ini juga bertujuan untuk mengetahui total pendapatan *sparepart* untuk perusahaan. Beberapa kolom yang ada pada laporan ini adalah tanggal transaksi, kode *sparepart*, nama *sparepart*, jumlah, harga, dan total harga. Desain laporan dapat dilihat pada gambar 4.21.

**Kawasaki** PT. Surapita Unitrans Surabaya

**Laporan Pendapatan Sparepart**

Tanggal : 31-03-2015 s/d 31-03-2015

No.	Tanggal Transaksi	Kode Sparepart	Nama Sparepart	Jumlah	Harga	Total
1	2015-03-31	16021-1203	SCREW-THROTTLE STOP	1	58,000	58,000
2	2015-03-31	13159-1056	CONNECTOR	1	11,000	11,000
3	2015-03-31	92049-0131	OIL	1	150,000	150,000
4	2015-03-31	16021-1203	SCREW-THROTTLE STOP	1	58,000	58,000
5	2015-03-31	13159-1056	CONNECTOR	1	11,000	11,000
Total Harga :						288,000

Cetak Printer    Close

**Gambar 4.21** Laporan Pendapatan Sparepart

#### D.4. Laporan Pekerjaan *Mechanic*

Laporan pendapatan *mechanic* ini digunakan untuk dokumentasi pekerjaan *mechanic* per periode. Selain itu laporan ini juga bertujuan untuk hasil pekerjaan *mechanic* yang telah dilakukan. Beberapa kolom yang ada pada laporan ini adalah *type* kendaraan, nopol, tanggal, nama pelanggan, jasa service & harga. Desain laporan dapat dilihat pada gambar 4.22.

**Kawasaki** PT. Surapita Unitrans Surabaya

**Laporan Pekerjaan Mekanik**

Mekanik : Hari Panca Tanggal : 18-08-2015 s/d 18-08-2015

No.	Type Kendaraan	Nopol	Tanggal	Nama Pelanggan	Jasa Service & Harga	
1	ER-6n	L4251UL	18-08-2015	Yulianto E Prabowo	<u>Jasa Service</u> Service berkala / Tune up Ongkos ganti oli	<u>Harga Service</u> 300,000 36,000
Total Harga :					336,000	

Dibuat Oleh Mengetahui

Admin Kepala Divisi After Sale

**Note :**

- Laporan untuk pekerjaan mechanic.

**Gambar 4.21** Laporan Pekerjaan *Mechanic*

### 4.3 Evaluasi Sistem

Setelah implementasi proses aplikasi pelayanan administrasi perawatan sepeda motor, maka langkah selanjutnya adalah membuat evaluasi atau testing sistem, agar dapat mengetahui kesesuaian sistem berjalan sesuai prosedur atau tidak dan dapat memastikan sistem terhindar dari *error*. *Testing* juga dapat digunakan untuk memastikan apakah data tersebut valid atau tidak dalam proses *input*, sehingga dapat menghasilkan *output* yang sesuai.

### 4.3.1 Uji Coba Form

Uji coba form ini berfokus pada pengujian sistem terhadap *handling error* pada setiap *inputan user*. Sebuah mekanisme yang digunakan untuk uji coba form ini adalah *test case*, yang mana menentukan apakah sebuah perangkat lunak atau sistem telah berhasil atau tidak dalam testing atau uji coba

#### A. Hasil Uji Coba Form Login

Pada saat program mulai dijalankan, form *login* akan tampil. Form ini digunakan untuk mengisi *username* dan *password* yang hanya bisa diakses oleh pihak yang bersangkutan untuk mengakses menu sesuai hak akses. *Form login* terlihat pada Gambar 4.23, sedangkan data *login* yang digunakan terlihat pada Tabel 4.1.

**Tabel 4.1** Test Case User Login

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Memastikan <i>user</i> yang memiliki hak akses saja yang dapat mengakses system	<i>Username</i> dan <i>password</i> yang tidak memiliki hak akses	Pengguna yang memiliki hak akses saja yang dapat masuk sistem.	<i>User</i> (pelanggan), kasir, admin dan kepala divisi saja yang dapat masuk sistem.	Sukses	Gambar 4.23
2	<i>Login</i> sesuai hak akses ( <i>Authentication User Privileges</i> )	Memasukkan <i>username</i> dan <i>password</i> (untuk hak akses karyawan, pelanggan dan kepala divisi)	Pengguna mengakses sistem sesuai <i>privileges</i> nya	Sistem diakses oleh pengguna sesuai <i>privileges</i> nya	Sukses	Gambar 4.24, 4.25, 4.26, 4.27.

*Test Case* Tabel 4.1 pada nomor 1 akan menampilkan *Form Login* yang *inputan* yang tidak memiliki hak akses dan ditampilkan pada Gambar 4.23.

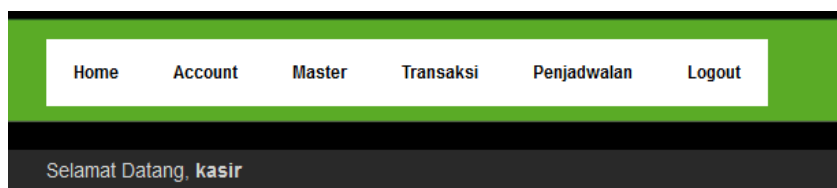
**Gambar 4.23** Tampilan Menu *Login*

Validasi pada gambar 4.23 diatas merupakan jendela peringatan yang muncul pada saat input *username* yang salah. Form utama akan muncul setelah *login* berhasil dan menu-menu yang aktif berdasarkan levelnya.

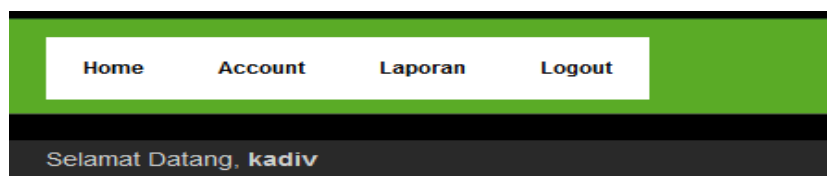
*Test Case* Tabel 4.2 pada nomor 2 akan menampilkan *Form Utama* setelah sukses pada form *login*. Dimana menu utama tersebut menyesuaikan dengan hak akses pengguna dan ditampilkan pada Gambar 4.24, 4.25, 4.26, dan 4.26.



**Gambar 4.24** Menu Utama *Admin*

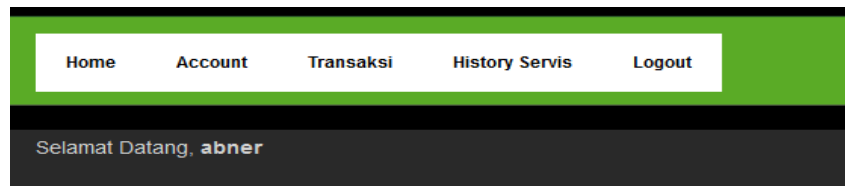


**Gambar 4.25** Menu Utama *Kasir*



**Gambar 4.26** Menu Utama *Kepala Divisi*





**Gambar 4.27** Halaman Utama Pelanggan

## B. Hasil Uji Coba Form *Master*

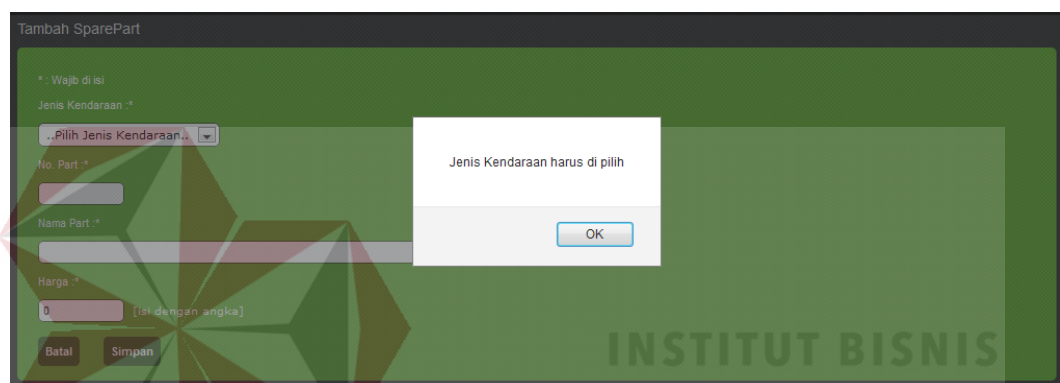
### B.1 Form Menu Master

Menu master mempunyai beberapa sub-sistem yaitu pendaftaran *user*, jenis kendaraan, kendaraan, *mechanic*, *sparepart*, hari operasional, perkiraan, dan detail *user*. Form ini digunakan untuk mengisi data master sepeda motor yang ada pada perusahaan.

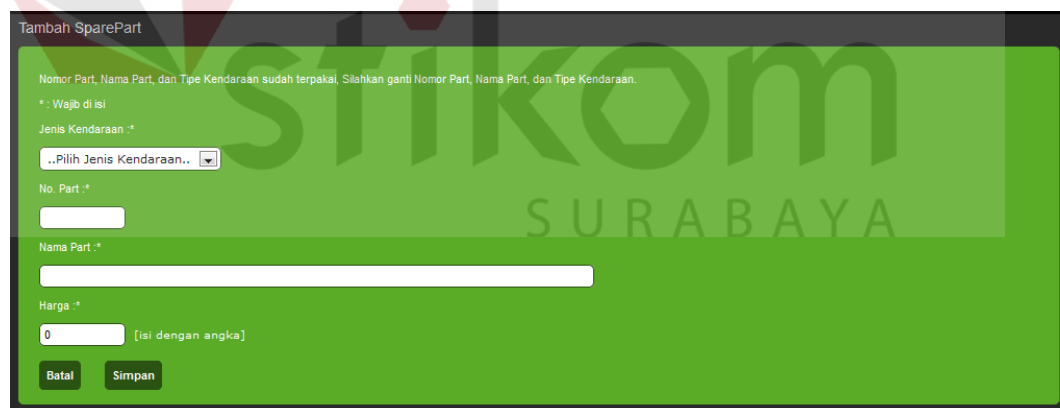
**Tabel 4.2** Test Case Menu Master

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Memastikan data diisi dengan valid dan lengkap (menu <i>sparepart</i> )	Pengguna tidak mengisi form <i>sparepart</i> dengan lengkap dan valid	Muncul <i>Message box</i> dan validasi sistem	<i>Message box</i> dan validasi sistem muncul ketika form tidak diisi lengkap dan tidak sesuai dengan validasi <i>input</i> sistem	Sukses	Gambar 4.28
2	Memastikan data tidak ada inputan yang double (menu <i>sparepart</i> )	Pengguna melakukan input data yang sudah terdaftar di dalam database	Muncul peringatan dan validasi sistem	Peringatan dan validasi sistem muncul ketika form diisi dengan data yang sudah terdapat pada database	Sukses	Gambar 4.29

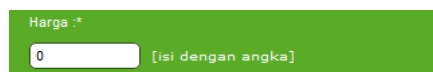
No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
3	Validasi input huruf dan angka (menu <i>sparepart</i> )	Input huruf pada <i>textbox</i>	Peringatan validasi inputan pengguna	tidak bisa melakukan <i>input</i> dengan huruf	Sukses	Gambar 4.30



Gambar 4.28 Message Box Form *Sparepart*



Gambar 4.29 Peringatan pada Form *Sparepart*



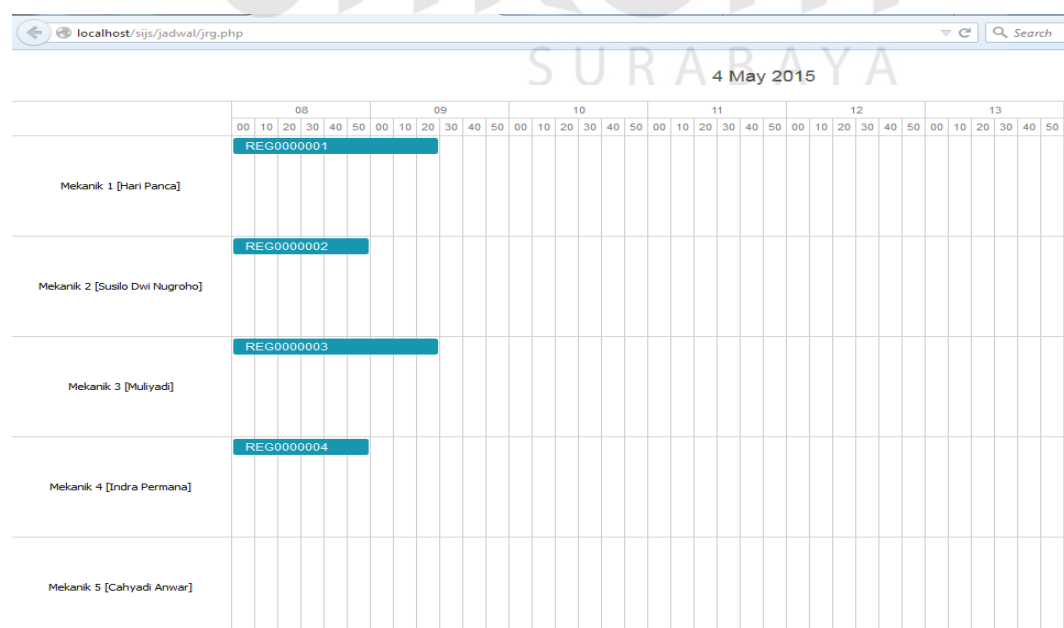
Gambar 4.30 Form Pemberitahuan pada *Sparepart*

### C. Hasil Uji Coba Form Menu Transaksi

Menu transaksi pada admin atau kasir mempunyai beberapa sub-sistem yaitu *reguler service*, *booking service*, *re-schedule booking service*, pemakaian *sparepart* dan pembayaran. Sedangkan menu transaksi pada pelanggan mempunyai sub-sistem *booking service*. Form ini digunakan untuk mengisi data transaksi sepeda motor sebelum melakukan *service*. *Input* data transaksi bisa diakses oleh admin, kasir dan pelanggan. Untuk melakukan uji coba pada form transaksi terdapat beberapa tahap uji coba antara lain uji coba *reguler*, *booking*, dan *reguler booking*.

#### C.1 Uji Coba Transaksi *Reguler*

Dalam transaksi ini uji coba yang dilakukan adalah transaksi *reguler*. Pelanggan yang ingin melakukan perawatan sepeda motor datang secara langsung ke perusahaan dan melakukan *registrasi service* secara langsung di perusahaan tersebut. Kondisi awal transaksi dapat dilihat pada gambar 4.31 dibawah ini.



Gambar 4.31 Penjadwalan *Reguler*

**Tabel 4.3** *Test Case* Transaksi Penjadwalan *Reguler*

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Memastikan data transaksi reguler sesuai dengan urutan antrian	No Polisi “W5442GL” nama “Abner” (otomatis antrian “REG0000005”)	Pelanggan masuk dalam sistem antrian sesuai dengan urutan antrian	Antrian dengan No Polisi “W5442GL” masuk dalam urutan <i>mechanic 5</i> dengan nama <i>mechanic</i> “Chayadi Anwar”	Sukses	Gambar 4.32 4.33
2	Memastikan data transaksi reguler sesuai dengan urutan antrian berdasarkan pendeknya antrian pada proses sebelumnya	No Polisi “W4521GH” nama “Doni Kiswanto” (otomatis antrian “REG0000006”)	Pelanggan masuk dalam sistem antrian sesuai dengan urutan antrian berdasarkan pendeknya sebuah proses	Antrian dengan No Polisi “W4521GH” masuk dalam urutan <i>mechanic 2</i> dengan nama <i>mechanic</i> “Susilo Dwi Nugroho”	Sukses	Gambar 4.34 4.35
3	Memastikan data transaksi tidak memiliki inputan data no polisi pada hari yang sama	No Polisi “W5442GL” nama “Abner”	Muncul peringatan dan validasi sistem	Peringatan dan validasi sistem muncul ketika form diisi dengan data no polisi pada hari yang sama	Sukses	Gambar 4.36
4	Memastikan penjadwalan yang melebihi jam <i>service</i> memiliki peringatan	No Polisi “L3123NY” Nama “Irene Liem”	Muncul peringatan dan validasi sistem	Peringatan dan validasi sistem muncul ketika perawatan yang dilakukan melebihi jam kerja <i>service</i>	Sukses	Gambar 4.37

*Test Case* tabel 4.3 pada nomor 1 akan ditampilkan pada gambar 4.32 sebagai hasil transaksi tampilan jadwal *reguler* dan gambar 4.33 sebagai detail data pelanggan yang akan melakukan transaksi *reguler*.

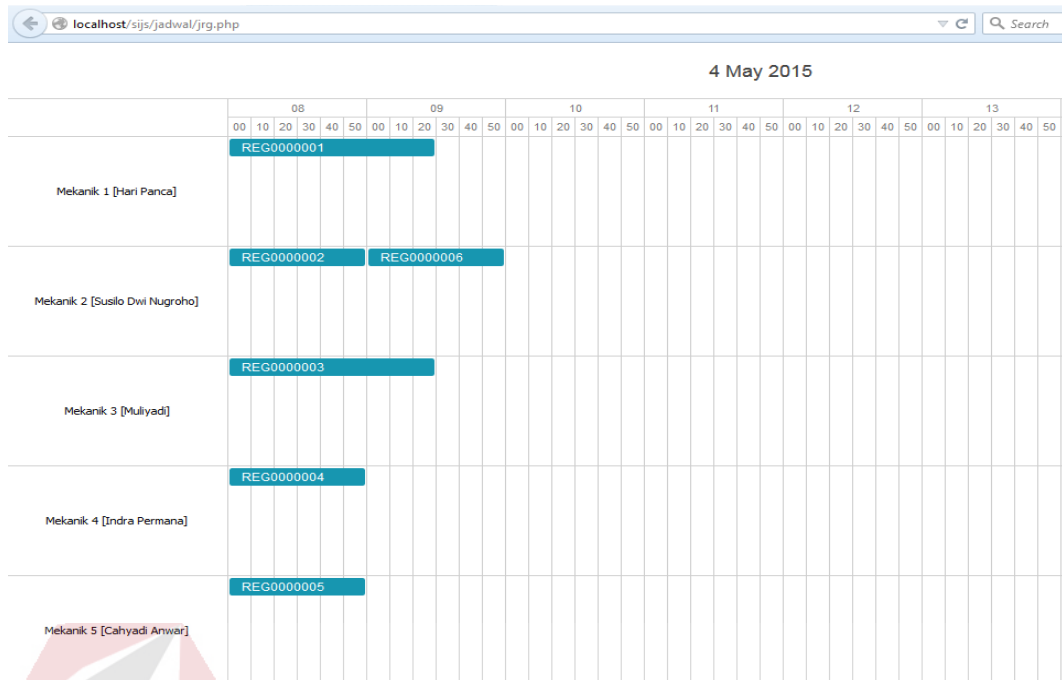
	08					09					10					11					12					13										
	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50
Mekanik 1 [Hari Panca]	REG0000001																																			
Mekanik 2 [Susilo Dwi Nugroho]	REG0000002																																			
Mekanik 3 [Mulyadi]	REG0000003																																			
Mekanik 4 [Indra Permana]	REG0000004																																			
Mekanik 5 [Cahyadi Anwar]	REG0000005																																			

**Gambar 4.32** Tampilan Transaksi *Reguler*

No.	Tanggal Registrasi	Kode Registrasi	Nomor Pelanggan	Nama Pelanggan	Nopol	Jam Durasi	Mekanik	Jumlah Estimasi Waktu	Jumlah Biaya	Status	Detail
1.	04-05-2015	REG0000005	KS0000001	Abner Mahendra	W5442GL	08:00:01 - 09:00:00	Cahyadi Anwar	60 menit	38.000	On Schedule & Job	
2.	04-05-2015	REG0000004	KS0000002	Herjono Arman	W2536KL	08:00:01 - 09:00:00	Indra Permana	60 menit	38.000	On Schedule & Job	
3.	04-05-2015	REG0000003	KS0000001	Abner Mahendra	L5211EW	08:00:01 - 09:30:00	Mulyadi	90 menit	70.000	On Schedule & Job	

**Gambar 4.33** Detail Transaksi *Reguler*

*Test Case* tabel 4.3 pada nomor 2 hasil dari tampilan tersebut akan ditampilkan pada gambar 4.34 sebagai hasil transaksi tampilan jadwal dan gambar 4.35 sebagai detail data pelanggan yang melakukan transaksi.



Gambar 4.34 Tampilan Transaksi *Reguler* berdasarkan pendeknya proses

No.	Tanggal Registrasi	Kode Registrasi	Nomor Pelanggan	Nama Pelanggan	Nopol	Jam Durasi	Mekanik	Jumlah Estimasi Waktu	Jumlah Biaya	Status	Detail
1.	04-05-2015	REG0000006	KS00000003	Doni Kiswanto	W4521GH	09:00:01 - 10:00:00	Susilo Dwi Nugroho	60 ment	150,000	On Schedule & Job	
2.	04-05-2015	REG0000005	KS00000001	Abner Mahendra	W5442GL	08:00:01 - 09:00:00	Cahyadi Anwar	60 ment	38,000	On Schedule & Job	

Gambar 4.35 Detail Transaksi *Reguler* berdasarkan pendeknya proses

Test Case tabel 4.3 pada nomor 3 dan 4 hanya menampilkan gambar baik berupa peringatan ataupun combobos yang ada pada gambar 4.36 dan 4.37

Nomor Polisi Kendaraan ini sudah teregistrasi hari ini.!

Masukkan Nopol Kendaraan :

Batal Proses

Peringatan muncul ketika no polisi sudah melakukan perawatan pada hari yang sama

Gambar 4.36 Input Nopol Transaksi *Reguler* pada hari yang sama

The screenshot shows the 'Tambah Regular Servis' form in the Kawasaki system. The form fields are: Tanggal (2015-04-08), Nomor Registrasi (REG0000007), Nomor Pelanggan (KS00000017), Name Pelanggan (Irene Lim), Nopol (L3123NY), and Type Kendaraan (Edge R). A red box highlights the error message: 'Ma'af, Transaksi Registrasi Servis untuk hari ini sudah penuh..!'. A red arrow points from this message to another red box containing the warning: 'Peringatan melebihi jam kerja'.

**Gambar 4.37** Peringatan melebihi jam kerja pada transaksi *Regular*

## C.2 Form Transaksi Penjadwalan *Booking*

Dalam transaksi ini uji coba tahap kedua yang dilakukan adalah transaksi *booking*. Dimana pelanggan yang ingin melakukan perawatan sepeda motor harus *login* terlebih dahulu kedalam sistem, setelah itu pelanggan bisa memilih jam dan hari sesuai keinginan pelanggan dengan cara melakukan *booking*. Kondisi awal transaksi pada transaksi *booking* dapat dilihat pada gambar 4.38 dibawah ini.

The screenshot shows the booking scheduling interface for 5 May 2015. The interface displays a grid of time slots (00, 10, 20, 30, 40, 50) for five mechanics. The bookings are as follows:

Mechanic	08	09	10	11	12	13
Mekanik 1 [Hari Panca]			BOK0000001			
Mekanik 2 [Susilo Dwi Nugroho]			BOK0000002			
Mekanik 3 [Mulyadi]					BOK0000003	
Mekanik 4 [Indra Permana]			BOK0000004			
Mekanik 5 [Cahyadi Anwar]						

**Gambar 4.38** Penjadwalan *Booking*

**Tabel 4.4** Test Case Transaksi Penjadwalan *Booking*

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Memastikan data transaksi <i>booking</i> sesuai dengan urutan antrian berdasarkan jam dan hari pilihan pelanggan	No Polisi "S4444MS" Tanggall "05-05-2015" Jam "10:00" Nama "Samuel Shiba" (otomatis antrian "BOK0000005")	Pelanggan masuk dalam sistem antrian sesuai dengan urutan antrian berdasarkan jam dan hari pilihan pelanggan	<i>Booking</i> dengan No Polisi "S4444MS" masuk dalam urutan <i>mechanic</i> 5 dengan nama <i>mechanic</i> "Chayadi Anwar"	Sukses	Gambar 4.39 4.40
2	Memastikan data transaksi <i>booking</i> masuk dalam slot pekerjaan <i>mechanic</i> berdasarkan slot ketersediaan <i>mechanic</i> pada jam yang ditentukan	No Polisi "M3221LO" Tanggall "05-05-2015" Jam "10:00" nama "Winny Merry Kalmante" (otomatis antrian "BOK0000006")	Pelanggan masuk dalam sistem antrian secara komputerisasi sesuai dengan slot ketersediaan <i>mechanic</i>	<i>Booking</i> dengan No Polisi "M3221LO" masuk dalam urutan <i>mechanic</i> 3 dengan nama <i>mechanic</i> "Mulyadi"	Sukses	Gambar 4.41 4.42
3	Memastikan data transaksi tidak memiliki inputan data no polisi pada hari yang sama	No Polisi "S4444MS" nama "Abner"	Muncul peringatan dan validasi sistem	Peringatan dan validasi sistem muncul ketika form diisi dengan data no polisi pada hari yang sama	Sukses	Gambar 4.43
4	Memastikan jika slot pengerjaan <i>mechanic</i> penuh memiliki peringatan	No Polisi "L3123NY" Nama "Irene Liem"	Muncul peringatan dan validasi sistem	Peringatan dan validasi sistem muncul ketika slot <i>mechanic</i> penuh	Sukses	Gambar 4.44



*Test Case* tabel 4.4 pada nomor 1 akan ditampilkan pada gambar 4.39 sebagai hasil transaksi tampilan jadwal *booking* dan gambar 4.40 sebagai detail data pelanggan yang akan melakukan transaksi *booking*.

	08					09					10					11					12					13				
	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50
Mekanik 1 [Hari Panca]													BOK0000001																	
Mekanik 2 [Susilo Dwi Nugroho]													BOK0000002																	
Mekanik 3 [Mulyadi]																									BOK0000003					
Mekanik 4 [Indra Permana]													BOK0000004																	
Mekanik 5 [Cahyadi Anwar]													BOK0000005																	

**Gambar 4.39** Tampilan Transaksi *Booking*

Home Account Transaksi History Servis Logout

Kelola Booking Servis

Pencarian :  Find.. Refresh

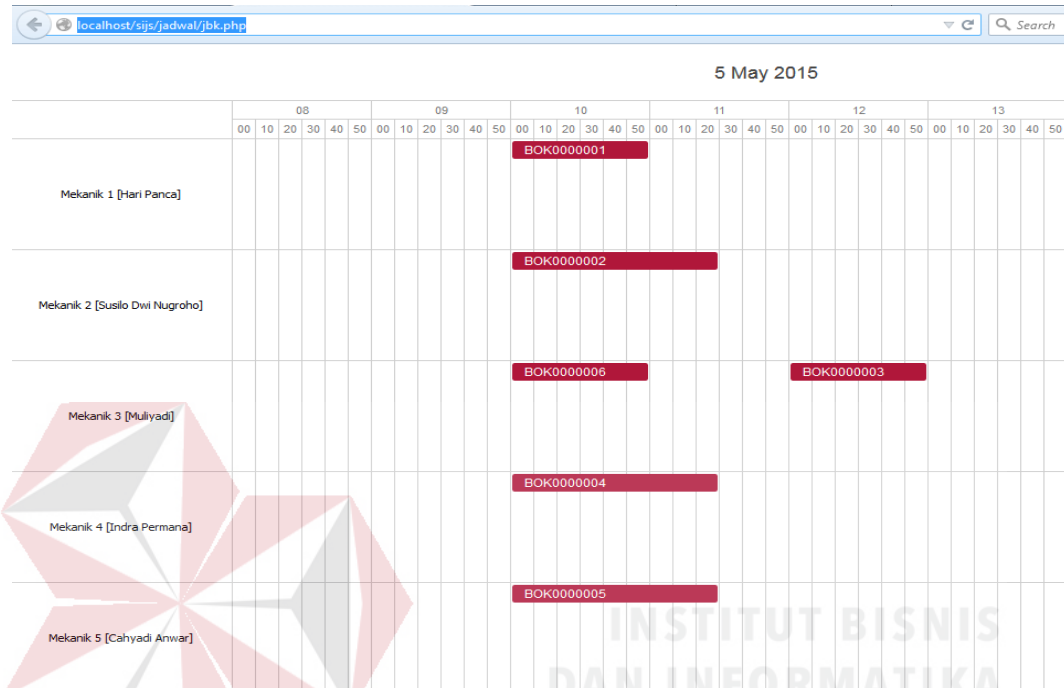
**Inputan booking sesuai dengan urutan pekerjaan mechanic**

No.	Tanggal Booking	Kode Booking	Nomor Pelanggan	Nama Pelanggan	Nopol	Jam Durasi	Mekanik	Jumlah Estimasi Waktu	Jumlah Biaya	Status	Detail
1.	05-05-2015	BOK0000005	KS00000005	Samuel Shiba	S4444MS	10:00:01 - 11:30:00	Cahyadi Anwar	90 menit	150,000	On Schedule & Job	

**Gambar 4.40** Detail Transaksi *Booking*

*Test Case* tabel 4.4 pada nomor 2 *input* data transaksi *booking* masuk dalam slot pekerjaan *mechanic* berdasarkan slot ketersediaan *mechanic* pada jam yang sama. hasil dari tampilan tersebut akan ditampilkan pada gambar 4.41

sebagai hasil transaksi tampilan jadwal dan gambar 4.42 sebagai detail data pelanggan yang melakukan transaksi.



**Gambar 4.41** Tampilan *Booking* berdasarkan ketersediaan *Mechanic*

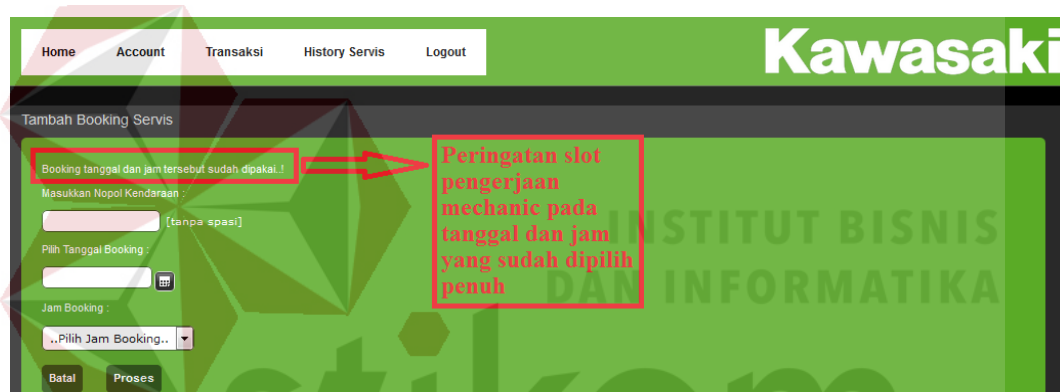
No.	Tanggal Booking	Kode Booking	Nomor Pelanggan	Nama Pelanggan	Nopol	Jam Durasi	Mekanik	Jumlah Estimasi Waktu	Jumlah Biaya	Status	Detail
1.	05-05-2015	BOK0000006	KS0000007	Winy Merry Kalmante	M3221LO	10:00:01 - 11:00:00	Mulyadi	60 menit	38,000	On Schedule & Job	

**Gambar 4.42** Detail *Booking* berdasarkan ketersediaan *Mechanic*

*Test Case* tabel 4.3 pada nomor 3 dan 4 hanya menampilkan gambar baik berupa peringatan ataupun combobos yang ada pada gambar 4.43 dan 4.44



**Gambar 4.43** Peringatan *Input* Nopol pada hari yang sama



**Gambar 4.44** Peringatan Slot ketersediaan *Mechanic*

### C.3 Uji Coba Transaksi *Reguler* dan *Booking*

Dalam transaksi ini uji coba tahap ketiga yang dilakukan adalah transaksi *reguler* dan *booking*. Dimana pelanggan yang ingin melakukan *booking* harus *login* terlebih dahulu untuk bisa memilih hari dan jam sesuai keinginan pelanggan. Setelah data diterima pada hari yang ditentukan oleh pelanggan *booking*, admin dapat melakukan *registrasi* untuk pelanggan *reguler* yang datang langsung ke perusahaan untuk dilakukan penjadwalan transaksi antara *booking* dan *reguler*. Untuk warna merah marun adalah posisi pelanggan yang melakukan *booking*, sedangkan yang berwarna biru muda adalah pelanggan yang datang langsung

tanpa melakukan *booking*. Kondisi awal transaksi dapat dilihat pada gambar 4.45 dibawah ini.

6 May 2015

	08					09					10					11					12					13				
	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50
Mekanik 1 [Hari Panca]																														
Mekanik 2 [Suslo Dwi Nugroho]																														
Mekanik 3 [Mulyadi]																														
Mekanik 4 [Indra Permana]																														
Mekanik 5 [Cahyadi Anwar]																														

**Gambar 4.45** Penjadwalan *Reguler* dan *Booking*

Tabel 4.5 Test Case Transaksi Penjadwalan *Reguler* dan *Booking*

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Memastikan data transaksi sesuai dengan urutan antrian	No Polisi “W2536KL” Nama “Herjono Arman” (otomatis antrian “BOK0000009”)	Pelanggan masuk dalam sistem antrian sesuai dengan urutan antrian	Antrian dengan No Polisi “W2536KL” masuk dalam urutan <i>mechanic</i> 5 dengan nama <i>mechanic</i> “Chayadi Anwar”	Sukses	Gambar 4.46 4.47
2	Memastikan data transaksi masuk dalam antrian berdasarkan prioritas pendeknya sebuah proses	No Polisi “W4521GH” nama “Doni Kiswanto” (otomatis antrian “REG0000010”)	Pelanggan masuk dalam sistem antrian secara komputerisasi berdasarkan prioritas pendeknya sebuah proses	Antrian dengan No Polisi “W4521GH” masuk dalam urutan <i>mechanic</i> 1 dengan nama <i>mechanic</i> “Hari Panca”	Sukses	Gambar 4.48 4.49
3	Memastikan data transaksi tidak memiliki inputan data no polisi pada hari yang sama	No Polisi “W2536KL” nama “Herjono Arman”	Muncul peringatan dan validasi sistem	Peringatan dan validasi sistem muncul ketika form diisi dengan data no polisi pada hari yang sama	Sukses	Gambar 4.50
4	Memastikan jika slot pengerjaan <i>mechanic</i> penuh memiliki peringatan	No Polisi “L3123NY” Nama “Irene Liem”	Muncul peringatan dan validasi sistem	Peringatan dan validasi sistem muncul ketika slot <i>mechanic</i> penuh	Sukses	Gambar 4.51

*Test Case* tabel 4.5 pada nomor 1 akan ditampilkan pada gambar 4.46 sebagai hasil transaksi tampilan jadwal *Reguler* dan *booking* dan gambar 4.47 sebagai detail data pelanggan yang akan melakukan transaksi *Reguler* dan *booking*.

	08					09					10					11					12					13										
	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50	00	10	20	30	40	50
Mekanik 1 [Hari Panca]																																				
Mekanik 2 [Susilo Dwi Nugroho]																																				
Mekanik 3 [Mulyadi]																																				
Mekanik 4 [Indra Permana]																																				
Mekanik 5 [Cahyadi Anwar]																																				

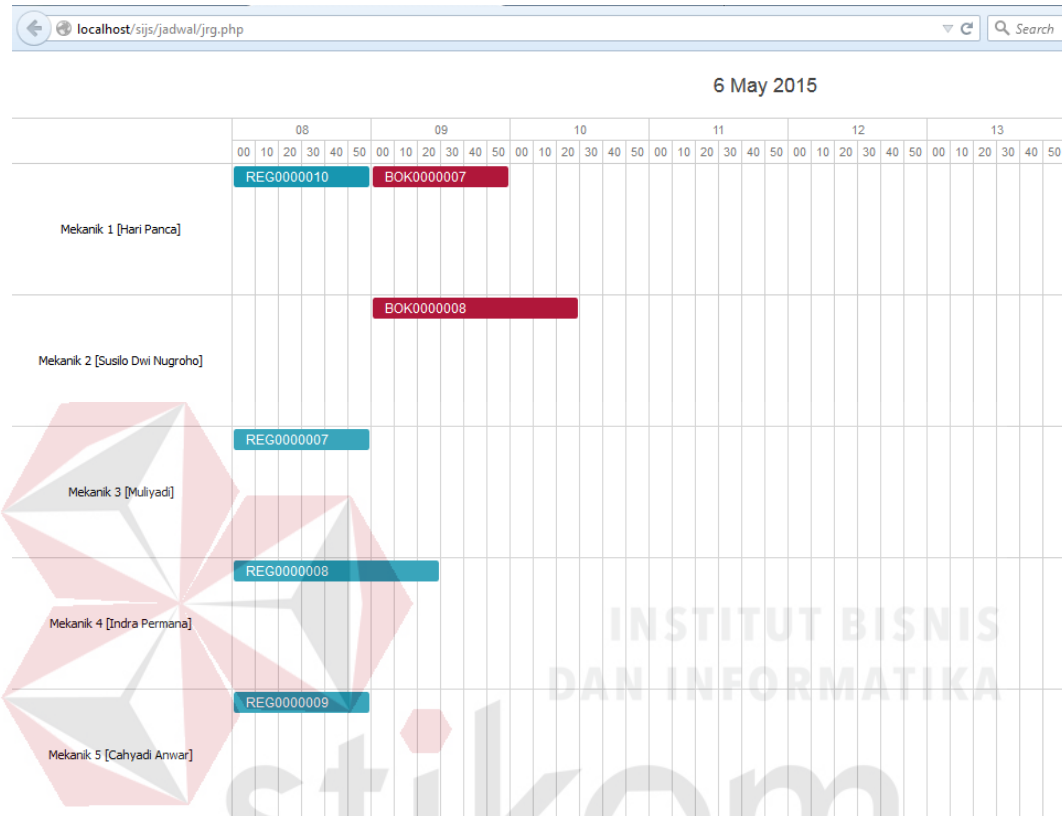
**Gambar 4.46** Tampilan Transaksi *Reguler* dan *Booking*

No.	Tanggal Registrasi	Kode Registrasi	Nomor Pelanggan	Nama Pelanggan	Nopol	Jam Durasi	Mekanik	Jumlah Estimasi Waktu	Jumlah Biaya	Status	Detail
1.	06-05-2015	REG0000009	KS00000002	Herjono Arman	W2536KL	08:00:01 - 09:00:00	Cahyadi Anwar	60 menit	38,000	On Schedule & Job	
2.	06-05-2015	REG0000008	KS00000002	Herjono Arman	W5623FF	08:00:01 - 09:30:00	Indra Permana	90 menit	300,000	On Schedule & Job	

**Gambar 4.47** Detail Transaksi *Reguler* dan *Booking*

*Test Case* tabel 4.4 pada nomor 2 menjelaskan data transaksi sesuai dengan urutan antrian berdasarkan prioritas pendeknya sebuah proses. Hasil dari

tampilan akan ditampilkan pada gambar 4.48 sebagai hasil transaksi tampilan jadwal *booking* dan gambar 4.49 sebagai detail data pelanggan yang akan melakukan transaksi *booking*.



**Gambar 4.48** Tampilan transaksi prioritas pendeknya proses

No.	Tanggal Registrasi	Kode Registrasi	Nomor Pelanggan	Nama Pelanggan	Nopol	Jam Durasi	Mekanik	Jumlah Estimasi Waktu	Jumlah Biaya	Status	Detail
1.	06-05-2015	REG0000010	KS00000003	Doni Kiswanto	W4521GH	08:00:01 - 09:00:00	Hari Panca	60 menit	150,000	On Schedule & Job	
2.	06-05-2015	REG0000009	KS00000002	Herjono Arman	W2536KL	08:00:01 - 09:00:00	Cahyadi Anwar	60 menit	38,000	On Schedule & Job	

**Gambar 4.49** Detail transaksi prioritas pendeknya proses

*Test Case* tabel 4.3 pada nomor 3 dan 4 hanya menampilkan gambar baik berupa peringatan ataupun combobos yang ada pada gambar 4.50 dan 4.51

The screenshot shows the 'Tambah Regular Servis' page. A red box highlights the warning message: 'Nomor Polisi Kendaraan ini sudah teregistrasi hari ini..!'. An arrow points from this message to another red box containing the text: 'Peringatan muncul ketika no polisi sudah melakukan perawatan pada hari yang sama'. Below the warning is a text input field for 'Masukkan Nopol Kendaraan : [tanpa spasi]' and two buttons: 'Batal' and 'Proses'.

**Gambar 4.50** Input Nopol Transaksi pada hari yang sama

The screenshot shows the 'Tambah Regular Servis' page with various input fields filled: Tanggal (2015-04-08), Nomor Registrasi (REG0000007), Nomor Pelanggan (KS00000017), Nama Pelanggan (Irene Lim), Nopol (L3123NY), and Type Kendaraan (Edge R). A red box highlights the warning message: 'Ma'af, Transaksi Registrasi Servis untuk hari ini sudah penuh..!'. An arrow points from this message to another red box containing the text: 'Peringatan melebihi jam kerja'. Below the warning are 'Batal' and 'Simpan' buttons.

**Gambar 4.51** Peringatan melebihi jam kerja

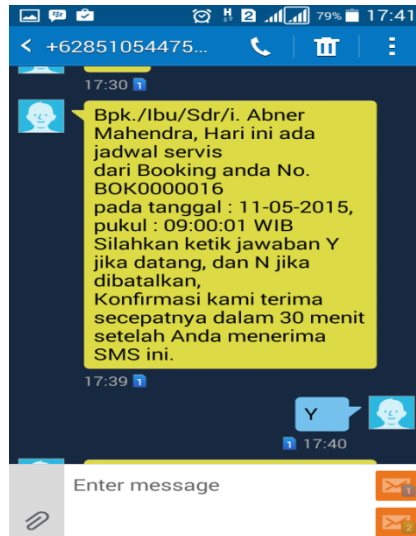
#### D. Hasil Uji Coba Sms Booking

Pelanggan akan menerima *sms booking* pada hari yang telah dipesan melalui layanan booking. *Sms booking* ini digunakan sebagai informasi kepada pelanggan untuk memastikan ketersediaan kehadiran pelanggan.

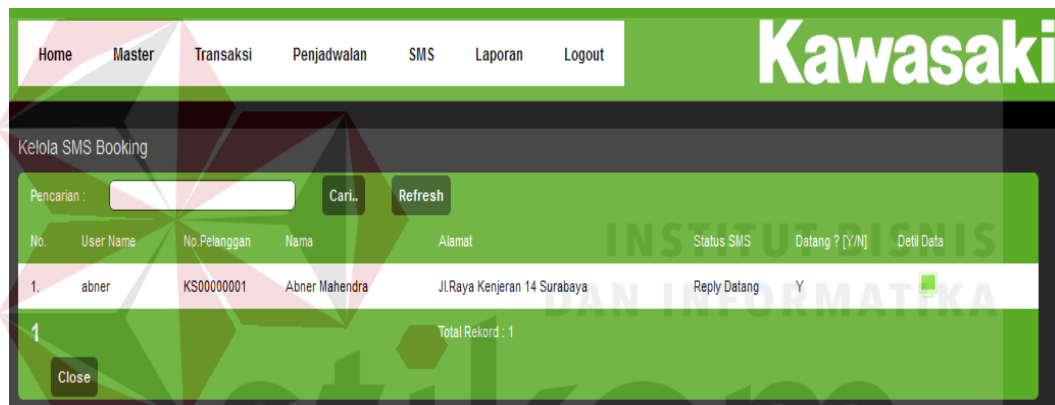
**Tabel 4.6** Test Case Sms Booking

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Memastikan ketersediaan kehadiran pelanggan	Ketik "Y"	Data <i>sms gateway</i> terkirim	Data <i>sms gateway</i> terkirim	Sukses	Gambar 4.52 4.53

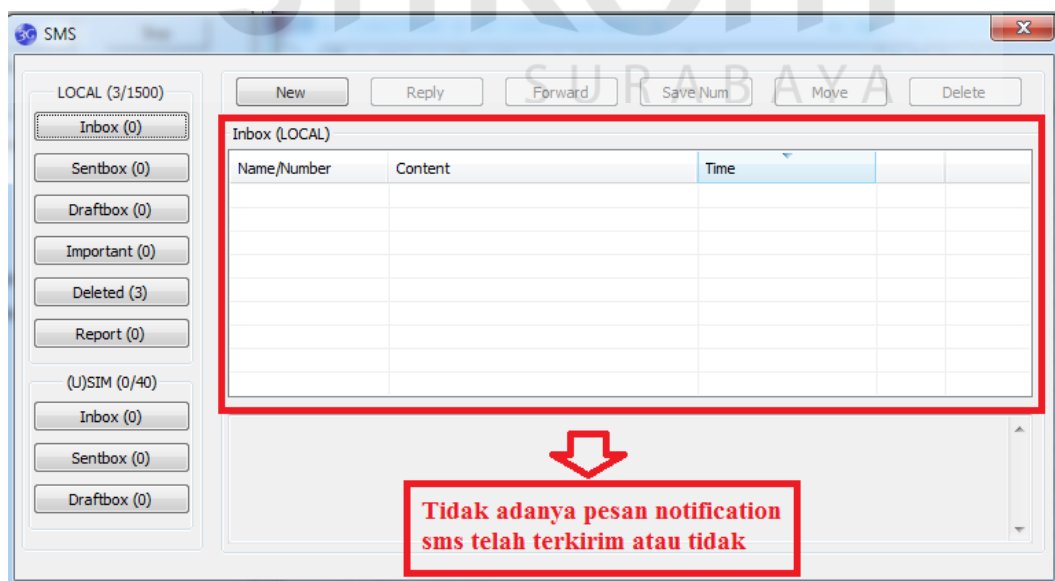




**Gambar 4.52** *Testing Sms Gateway*



**Gambar 4.53** *Detail Testing Sms Gateway*



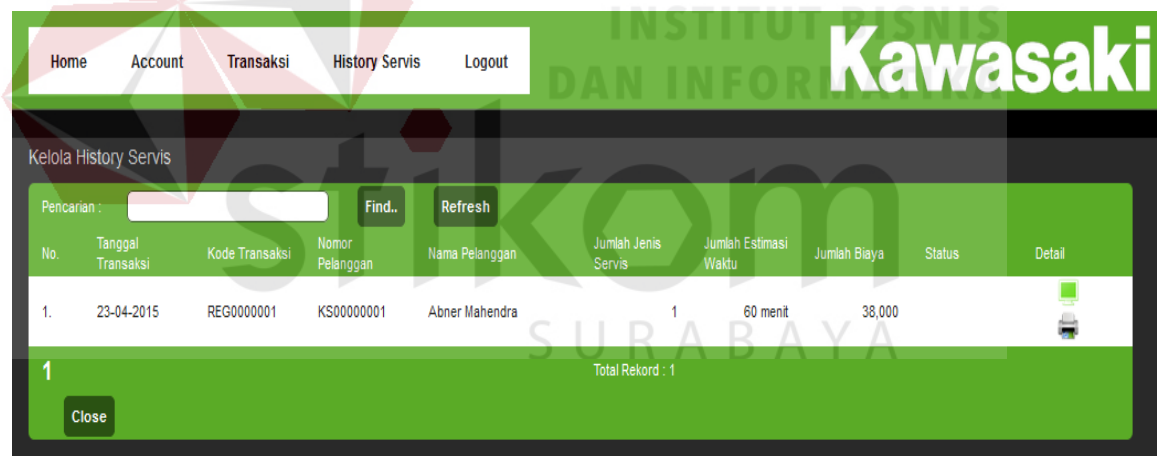
**Gambar 4.54** *Testing Sms Gateway tidak adanya notification*

### E. Hasil Uji Coba *History Service*

Menu *history service* dapat digunakan pelanggan.. Menu ini digunakan sebagai informasi kepada pelanggan untuk mengetahui *service* apa saja yang pernah dilakukan.

**Tabel 4.7** *Test Case History Service*

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Menampilkan data <i>history service</i>	Username “abner” Password “abner” (Memilih menu <i>history service</i> )	Data <i>history service</i> tampil pada <i>gridview</i>	Data <i>history service</i> tampil	Sukses	Gambar 4.55



**Gambar 4.55** *Testing History Service*

### F. Hasil Uji Coba Laporan

Menu laporan dapat digunakan admin dan kepala divisi. Laporan ini mempunyai beberapa sub-sistem yaitu laporan *service*, laporan kerusakan *service*, laporan pendapatan jasa, laporan pendapatan *sparepart*, laporan pekerjaan *mechanic*. Menu ini digunakan sebagai laporan layanan administrasi perawatan

sepeda motor kepada kepala divisi sesuai dengan kebutuhan perusahaan. Hasil laporan antara lain laporan *service*, laporan kerusakan *service*, laporan pendapatan jasa, laporan pendapatan *sparepart*, dan laporan pekerjaan *mechanic*.

### F.1 Laporan Service

Laporan *service* ini digunakan untuk mengetahui dokumentasi *service* secara keseluruhan pekerjaan *mechanic* berdasarkan pilihan *service* yang telah ditentukan perusahaan, serta adanya informasi biaya *service* dan *sparepart*.

**Tabel 4.8** Test Case Laporan Service

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Menampilkan data laporan <i>service</i>	Tanggal "04-05-2015" Sampai "05-05-2015"	Data laporan <i>service</i> tampil pada <i>gridview</i>	Data laporan <i>service</i> tampil	Sukses	Gambar 4.56

localhost/sjjs/admin/lp2\_ect.php

**Kawasaki** PT.Surapita Unitrans Surabaya

Laporan Service

Tanggal : 04-05-2015 s/d 05-05-2015

No.	Nama Mekanik	Ringan				Berat	Biaya	
		KSG	RTN	RGK	LTRK		Service	Sparepart
1	Hari Panca	0	2	0	0	0	338,000	208,000
2	Susito Dwi Nugroho	0	3	0	0	0	488,000	39,500
3	Mulyadi	0	3	0	0	0	338,000	6,500
4	Indra Permana	0	2	0	0	0	450,000	12,500
5	Cahyadi Anwar	0	2	0	0	0	188,000	58,000
Jumlah :		0	12	0	0	0	1,802,000	324,500

**Gambar 4.56** Testing Laporan Service

## F.2 Laporan Kerusakan Service

Laporan kerusakan *service* ini digunakan untuk mengetahui dokumentasi kerusakan *service* apa saja yang dilakukan per-periode. Selain itu laporan ini juga bertujuan untuk mengetahui nama pelanggan, tanggal *service*, tipe kendaraan yang melakukan perawatan sepeda motor.

**Tabel 4.9** Test Case Laporan Kerusakan Service

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Menampilkan data laporan kerusakan <i>service</i>	Tanggal "04-05-2015" Sampai "05-05-2015"	Data laporan kerusakan <i>service</i> tampil pada <i>gridview</i>	Data laporan kerusakan <i>service</i> tampil	Sukses	Gambar 4.57

No.	Tanggal Servis	Nama Pelanggan	Nomor Polisi	Tipe Kendaraan	Jasa Service	Mekanik
1	04-05-2015	Doni Kiswanto	W4521GH	Ninja RR Special Edition	- Service berkala / Tune up	Susilo Dwi Nugroho
2	04-05-2015	Abner Mahendra	W5442GL	Athlete	- Service berkala / Tune up	Cahyadi Anwar
3	04-05-2015	Wahyu Kurniawan	W4444YU	Ninja RR Special Edition	- Service berkala / Tune up	Indra Permana
4	04-05-2015	Samuel Shiba	S4444MS	Ninja 250	- Service berkala / Tune up	Muliyadi
5	04-05-2015	Winnie Merry Kalmante	M3221LO	Edge R	- Service berkala / Tune up	Susilo Dwi Nugroho
6	04-05-2015	Herjono Arman	W5623FF	ER-6n	- Service berkala / Tune up	Hari Panca
7	05-05-2015	Abner Mahendra	W5442GL	Athlete	- Service berkala / Tune up	Hari Panca
8	05-05-2015	Herjono Arman	W5623FF	ER-6n	- Service berkala / Tune up	Susilo Dwi Nugroho
9	05-05-2015	Doni Kiswanto	W4521GH	Ninja RR Special Edition	- Service berkala / Tune up	Muliyadi
10	05-05-2015	Joko Suryanto	L5412RT	ER-6n	- Service berkala / Tune up	Indra Permana
11	05-05-2015	Samuel Shiba	S4444MS	Ninja 250	- Service berkala / Tune up	Cahyadi Anwar
12	05-05-2015	Winnie Merry Kalmante	M3221LO	Edge R	- Service berkala / Tune up	Muliyadi

Total Data : 12

**Gambar 4.57** Testing Laporan Kerusakan Service

### F.3 Laporan Pendapatan Jasa

Laporan pendapatan jasa ini digunakan untuk mengetahui dokumentasi pendapatan jasa per-periode. Selain itu laporan ini juga bertujuan untuk mengetahui nama pelanggan, tanggal *service*, tipe kendaraan jasa, dan harga jasa.

**Tabel 4.10** *Test Case* Laporan Pendapatan Jasa

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Menampilkan data laporan pendapatan jasa	Tanggal "04-05-2015" Sampai "05-05-2015"	Data laporan pendapatan jasa tampil pada <i>gridview</i>	Data laporan pendapatan jasa tampil	Sukses	Gambar 4.58

localhost/sjs/admin/lp4\_act.php

**Kawasaki** PT.Surapita Unitrans Surabaya

Laporan Pendapatan Jasa Service

Tanggal : 04-05-2015 s/d 05-05-2015

No.	Tanggal Servis	Nama Pelanggan	Tipe Kendaraan	Jasa Service & Harga	
1	04-05-2015	Doni Kiswanto	Ninja RR Special Edition	<u>Jasa Service</u> Service berkala / Tune up	<u>Harga Service</u> 150,000
2	04-05-2015	Abner Mahendra	Athlete	<u>Jasa Service</u> Service berkala / Tune up	<u>Harga Service</u> 38,000
3	04-05-2015	Wahyu Kurniawan	Ninja RR Special Edition	<u>Jasa Service</u> Service berkala / Tune up	<u>Harga Service</u> 150,000
4	04-05-2015	Samuel Shiba	Ninja 250	<u>Jasa Service</u> Service berkala / Tune up	<u>Harga Service</u> 150,000
5	04-05-2015	Winnie Merry Kalmante	Edge R	<u>Jasa Service</u> Service berkala / Tune up	<u>Harga Service</u> 38,000
6	04-05-2015	Herjono Arman	ER-6n	<u>Jasa Service</u> Service berkala / Tune up	<u>Harga Service</u> 300,000
7	05-05-2015	Abner Mahendra	Athlete	<u>Jasa Service</u> Service berkala / Tune up	<u>Harga Service</u> 38,000
8	05-05-2015	Herjono Arman	ER-6n	<u>Jasa Service</u> Service berkala / Tune up	<u>Harga Service</u> 300,000

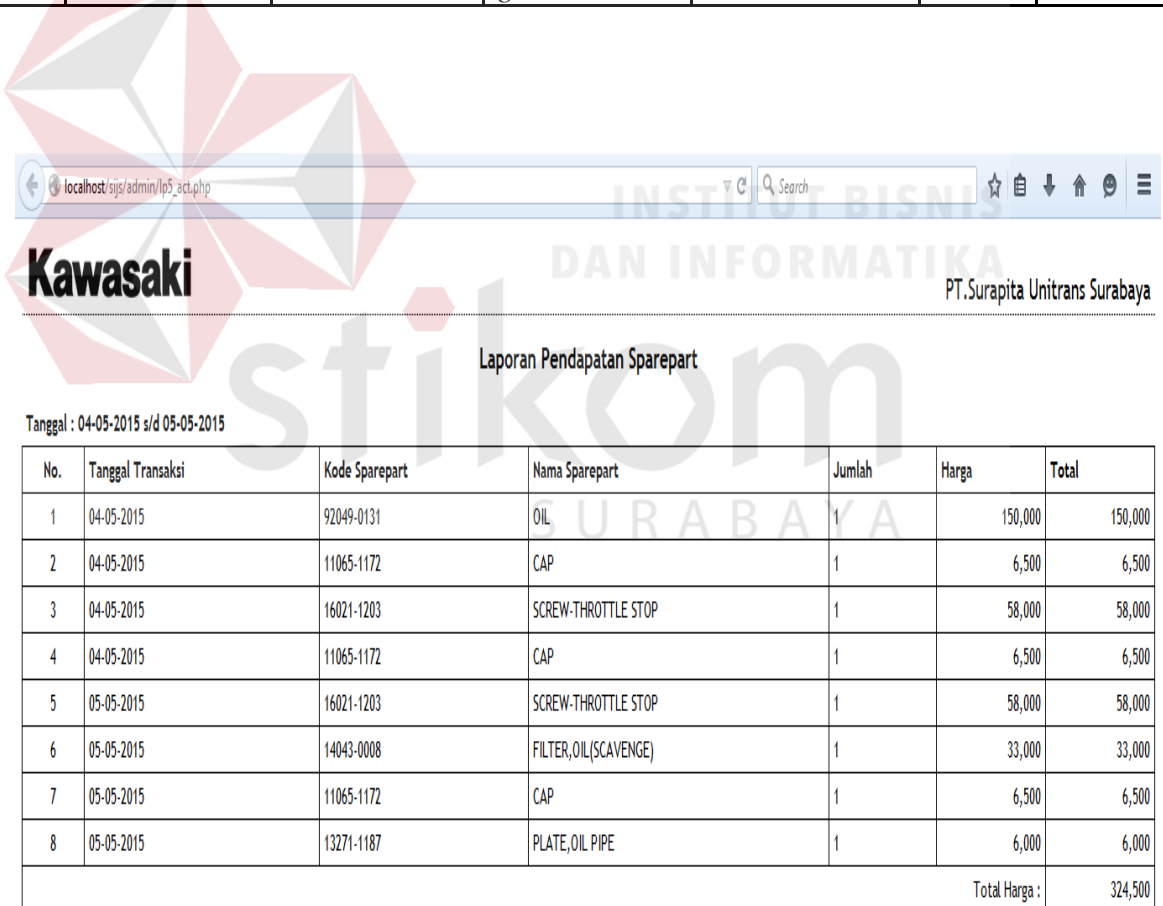
**Gambar 4.58** *Testing* Laporan Pendapatan Jasa Service

#### F.4 Laporan Pendapatan *Sparepart*

Laporan pendapatan *sparepart* ini digunakan untuk mengetahui dokumentasi pendapatan *sparepart* per-periode. Selain itu laporan ini juga bertujuan untuk mengetahui total pendapatan *sparepart* untuk perusahaan

**Tabel 4.11** *Test Case* Laporan Pendapatan *Sparepart*

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Menampilkan data laporan pendapatan <i>sparepart</i>	Tanggal "04-05-2015" Sampai "05-05-2015"	Data laporan pendapatan <i>sparepart</i> tampil pada <i>gridview</i>	Data laporan pendapatan <i>sparepart</i> tampil	Sukses	Gambar 4.59



The screenshot shows a web browser window with the URL `localhost/sjjs/admin/lp5_act.php`. The page header includes the logo for 'Kawasaki' and 'PT. Surapita Unitrans Surabaya'. The main content is a report titled 'Laporan Pendapatan Sparepart' for the period 'Tanggal : 04-05-2015 s/d 05-05-2015'. The report is presented as a table with the following data:

No.	Tanggal Transaksi	Kode Sparepart	Nama Sparepart	Jumlah	Harga	Total
1	04-05-2015	92049-0131	OIL	1	150,000	150,000
2	04-05-2015	11065-1172	CAP	1	6,500	6,500
3	04-05-2015	16021-1203	SCREW-THROTTLE STOP	1	58,000	58,000
4	04-05-2015	11065-1172	CAP	1	6,500	6,500
5	05-05-2015	16021-1203	SCREW-THROTTLE STOP	1	58,000	58,000
6	05-05-2015	14043-0008	FILTER,OIL(SCAVENGE)	1	33,000	33,000
7	05-05-2015	11065-1172	CAP	1	6,500	6,500
8	05-05-2015	13271-1187	PLATE,OIL PIPE	1	6,000	6,000
					Total Harga :	324,500

**Gambar 4.59** *Testing* Laporan Pendapatan *Sparepart*

## F.5 Laporan Pekerjaan *Mechanic*

Laporan pekerjaan *mechanic* digunakan untuk mengetahui dokumentasi keseluruhan pekerjaan yang telah dilakukan *mechanic* berdasarkan kebutuhan perusahaan

**Tabel 4.12** *Test Case* Laporan Pekerjaan *Mechanic*

No	Tujuan	Input	Output yang Diharapkan	Output Sistem	Status	Output
1	Menampilkan data laporan pekerjaan <i>Mechanic</i>	Tanggal "04-05-2015" Sampai "05-05-2015"	Data laporan pekerjaan <i>Mechanic</i> tampil pada <i>gridview</i>	Data laporan pekerjaan <i>Mechanic</i> tampil	Sukses	Gambar 4.60

**Kawasaki** PT. Surapita Unitrans Surabaya

Laporan Pekerjaan Mekanik

Tanggal : 04-05-2015 s/d 05-05-2015

Mekanik : Hari Panca

No.	Type Kendaraan	Nopol	Tanggal	Nama Pelanggan	Jasa Service & Harga
1	ER-6n	W5623FF	04-05-2015	Herjono Arman	<u>Jasa Service</u> Service berkala / Tune up <u>Harga Service</u> 300,000
2	Athlete	W5442GL	05-05-2015	Abner Mahendra	<u>Jasa Service</u> Service berkala / Tune up <u>Harga Service</u> 38,000
Total Harga :					338,000

Mekanik : Susilo Dwi Nugroho

No.	Type Kendaraan	Nopol	Tanggal	Nama Pelanggan	Jasa Service & Harga
1	Ninja RR Special Edition	W4521GH	04-05-2015	Doni Kiswanto	<u>Jasa Service</u> Service berkala / Tune up <u>Harga Service</u> 150,000
2	Edge R	M3221LO	04-05-2015	Winnie Merry Kalmanate	<u>Jasa Service</u> Service berkala / Tune up <u>Harga Service</u> 38,000
3	ER-6n	W5623FF	05-05-2015	Herjono Arman	<u>Jasa Service</u> Service berkala / Tune up <u>Harga Service</u> 300,000
Total Harga :					826,000

**Gambar 4.60** *Testing* Laporan Pekerjaan *Mechanic*

### 4.3.2 Hasil Evaluasi

Berdasarkan uji coba yang telah dilakukan seperti diatas, terdapat beberapa penjelasan uji coba yang yang berdasarkan latar belakang masalah, antara lain:

- Pada transaksi dilakukan dengan 3 cara uji coba antara lain *reguler*, *booking*, dan *reguler dan booking*
- Untuk *sms booking* terdapat konfirmasi ketersediaan apakah pelanggan hadir atau tidak
- Menampilkan *history service* untuk pelanggan
- Menampilkan laporan yang dipilih oleh kepala divisi apakah sesuai atau tidak

Berdasarkan penjelasan diatas maka dibuat tabel pengujian berdasarkan presentasi untuk memperoleh hasil evaluasi seperti pada tabel dibawah ini :

**Tabel 4.13** Tabel Pengujian Form Transaksi

No	Uji Coba	Berhasil	Tidak Berhasil	1%-100%
1	Transaksi Reguler	6 kali inputan dengan hasil 100% (Halaman 117 – 120 gambar 4.31, 4.32, 4.34)	Saat melebihi jam kerja peringatan keluar di button simpan (gambar 4.61)	86%
2	Transaksi Booking	6 kali inputan dengan hasil 100% (Halaman 121 – 124 gambar 4.38, 4.39, 4.41)	Saat melebihi jam kerja peringatan keluar di button simpan (gambar 4.61)	86%
3	Transaksi Reguler dan Booking	6 kali inputan dengan hasil 100% (Halaman 126 – 129 gambar 4.45, 4.46, 4.48)	Saat melebihi jam kerja peringatan keluar di button simpan (gambar 4.61)	86%



Gambar 4.61 Testing Interface Transaksi

Tabel 4.14 Tabel Pengujian sms booking

No	Uji Coba	Berhasil	Tidak Berhasil	1%-100%
1	Sms booking	Data sms gateway terkirim dan tersimpan dalam database dengan membalas sms ketik : "Y" (halaman 129 gambar 4.51)	Data terkirim atau tidaknya tidak memiliki notification (halaman 130 gambar 4.53)	50%

Tabel 4.15 Tabel Pengujian History Service

No	Uji Coba	Input	Hasil Output	Berhasil	Tidak Berhasil	1%-100%
1	Menampilkan History Service	Username "abner" Password "abner" (Memilih menu history service)	Gambar 4.52 Halaman 130	1	0	100%

Tabel 4.16 Tabel Pengujian Laporan

No	Uji Coba	Input	Hasil Output	Berhasil	Tidak Berhasil	1%-100%
1	Menampilkan data laporan	Tanggal "04-05-2015" Sampai "05-05-2015"	Gambar 4.57 Halaman 136	5	0	100%

Berdasarkan pada tabel pengujian yang telah dilakukan seperti tabel 4.13, 4.14, 4.15, 4.16. Berdasarkan presentasi maka diperoleh hasil dari evaluasi sebagai berikut :

- Informasi tentang antrian *service* pelanggan sudah sesuai dengan desain
- Pelanggan bisa melakukan *booking service* dan bisa melakukan konfirmasi ketersediaan hadir atau tidak tanpa memiliki berita *notification* terkirim atau tidaknya sms tersebut.
- Pelanggan mendapatkan informasi berupa *history service*.
- Terdapat laporan bagi pihak kepala divisi *after sale* berupa informasi.



## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Kesimpulan yang dapat diambil dari pembuatan Rancang Bangun Aplikasi *Administrasi* Pelayanan Perawatan Sepeda Motor di PT. Surapita Unitrans Surabaya dan evaluasi sistem adalah sebagai berikut:

1. Informasi tentang antrian *service* pelanggan sudah sesuai dengan desain.
2. Pelanggan bisa melakukan *booking service* dan bisa melakukan konfirmasi ketersediaan hadir atau tidak tanpa memiliki berita *notification* terkirim atau tidaknya sms tersebut.
3. Pelanggan mendapatkan informasi berupa *history service*.
4. Terdapat laporan bagi pihak kepala divisi *after sale* berupa informasi antara lain *service* motor pelanggan, data master, catatan kerusakan, pendapatan jasa, pendapatan *sparepart*, laporan pekerjaan *mechanic*.

#### **5.2 Saran**

Berdasarkan aplikasi yang telah dibuat, saran yang dapat disampaikan oleh penulis untuk pengembangan Rancang Bangun Aplikasi *Administrasi* Perawatan Sepeda Motor adalah sistem ini dapat dikembangkan dengan membuat aplikasi yang berhubungan dengan penjualan misal: aplikasi penjualan *sparepart* online.

## DAFTAR PUSTAKA

- Anhar. 2010. *Panduan Menguasai PHP & MySQL secara Otodidak. Cetakan Pertama*. Jakarta : Media Kita.
- Barata, A.A. 2008. *Dasar – Dasar Pelayanan Prima : Persiapan Membangun Budaya Pelayanan Prima untuk Meningkatkan Kepuasan dan Loyalitas Pelanggan*. Jakarta : Elex Media Komputindo.
- Hariningsih, HP. 2003. *Sistem Operasi*. Yogyakarta : Graha Ilmu.
- Indira dan Prianka, 2004, *GSM System-Indonesia*, 11 Desember 2013,  
[URL:http://www.indira.com/mobile-system/GSM-System-Indonesia.pdf](http://www.indira.com/mobile-system/GSM-System-Indonesia.pdf)
- Jogiyanto, H.M. 1999. *Analisa dan Desain Sistem Informasi*. Andi : Yogyakarta.
- Kendall, dan Kendall. 2003. *Analisis dan Perancangan Sistem Jilid 1*. Jakarta : Prenhallindo.
- Kustiyaningsih, Y. 2011. *Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL*. Yogyakarta : Graha Ilmu.
- Marlinda, Linda S.Kom. 2004. *Sistem Basis Data*, Andi Offset, Yogyakarta
- Nugroho, B. 2013. *Dasar Pemrograman Web PHP – MySQL dengan Dreamweaver (Study Kasus : Sistem Penerimaan Siswa Baru (PSB) Online)*. Cetakan I. Yogyakarta : Gava Media.
- Nuraida, I. 2008. *Manajemen Administrasi Perkantoran*. Yogyakarta : Kanisius.
- Pressman, R. S. 2001. *Rekayasa Perangkat Lunak : Pendekatan Praktisi, Edisi ke 1*. Yogyakarta:Andi
- Sakur, S. 2010. *PHP 5 Pemograman Berorientasi Objek : Konsep dan Implementasi*. Yogyakarta : ANDI.
- Tarigan, D. 2013. *Membangun SMS Gateway Berbasis WEB dengan Codeigniter*. Yogyakarta : Lokomedia.
- Yoeti. O. 1999. *Customer Service Cara Efektif Memuaskan Pelanggan*. Jakarta : PT. Pradnya Paramita.