

RANCANG BANGUN *HELPDESK TICKETING SYSTEM*

(STUDI KASUS: PT. PRIMUS INDOJAYA)



Oleh:

QOYYIMAH

204093002662

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH

JAKARTA

2011 M/1432 H

RANCANG BANGUN *HELPDESK TICKETING SYSTEM*

(STUDI KASUS: PT. PRIMUS INDOJAYA)

Skripsi

Sebagai Salah Satu Syarat Untuk Memperoleh Gelar

Sarjana Komputer

Pada Fakultas Sains dan Teknologi

Universitas Islam Negeri Syarif Hidayatullah Jakarta

Oleh:

QOYYIMAH

204093002662

Menyetujui,

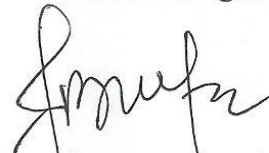
Pembimbing I



Nur Aeni Hidayah, MMSI

NIP. 19750818 200501 2 008

Pembimbing II



Zainuddin Bey Fananie, MSc

NIP.

Mengetahui,

Ketua Program Studi Sistem Informasi



Nur Aeni Hidayah, MMSI

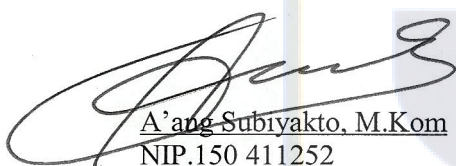
NIP. 19750818 200501 2 008

PENGESAHAN UJIAN


Skripsi berjudul “**Rancang Bangun Helpdesk Ticketing System (Studi Kasus: PT. Primus Indojaya)**” yang ditulis oleh Qoyyimah, NIM 204093002662 telah diuji dan dinyatakan lulus dalam sidang Munaqosyah Fakultas Sains dan Teknologi, Universitas Islam Negeri Syarif Hidayatullah Jakarta pada tanggal 29 Juli 2011. Skripsi ini telah diterima sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu (S1) Program Studi Sistem Informasi.

Menyetujui,

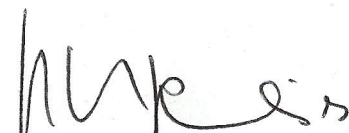
Penguji I


A'ang Subiyakto, M.Kom
NIP.150 411252

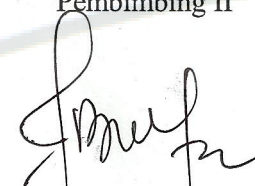
Penguji II


Nia Kumaladewi, MMSI
NIP.197504122007102002

Pembimbing I


Nur Aeni Hidayah, MMSI
NIP. 19750818 200501 2 008

Pembimbing II



Zainuddin Bey Fananie, M.Sc
NIP.

Mengetahui :

Dekan
Fakultas Sains dan Teknologi


Dr. Syopiansyah Jaya Putra, M.Sis
NIP.19680117 200112 1 001

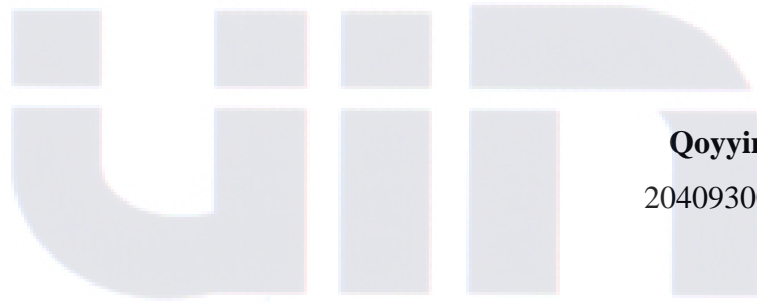
Ketua
Program Studi Sistem Informasi


Nur Aeni Hidayah, MMSi
NIP. 19750818 200501 2 008

PERNYATAAN

DENGAN INI SAYA MENYATAKAN BAHWA SKRIPSI INI BENAR-BENAR HASIL KARYA SENDIRI YANG BELUM PERNAH DIAJUKAN SEBAGAI SKRIPSI ATAU KARYA ILMIAH PADA PERGURUAN TINGGI ATAU LEMBAGA MANAPUN.

Jakarta, Juli 2011



Qoyyimah

204093002662

ABSTRAK

QOYYIMAH (204093002662). Rancang Bangun *Helpdesk Ticketing System* Studi Kasus: PT. Primus Indojoya. di bawah bimbingan **NUR AENI HIDAYAH** dan **ZAINUDDIN BEY FANANIE**.

PT. Primus Indojoya merupakan sebuah perusahaan yang terintegrasi, terdistribusi, dan terlengkap dalam memberikan informasi mengenai teknologi informasi. *Helpdesk ticketing system* yang terkomputerisasi sangat mendesak diperlukan oleh PT. Primus Indojoya, dikarenakan dalam menjalankan proses usaha perusahaan seringkali ditemukan masalah-masalah pada peralatan penunjang yang ada dilapangan dan harus segera diselesaikan. Adapun masalah yang berkaitan adalah tentang *hardware* dan *software*. Namun penanganan masalah yang saat ini terjadi pada *helpdesk ticketing system* di PT. Primus Indojoya masih kurang baik, karena proses pencatatan *helpdesk* yang masih *manual* dan sederhana sehingga mempengaruhi kinerja perusahaan tidak cepat teratasi, dan prosesnya tidak terkontrol. Semua pencatatan laporan bulanan serta perhitungan masalah masih dilakukan *manual*. Hal ini dapat memberikan dampak negatif, yakni: terdapat masalah yang dicatat secara berulang, perhitungan masalah tidak akurat karena dilakukan manual dan perlunya manajer operasional menunggu untuk mendapatkan laporan yang diinginkan. Dan juga belum adanya sistem yang mampu mengukur tingkat masalah pada *helpdesk* dalam menangani permasalahan yang terjadi pada PT. Primus Indojoya. Oleh karena itu dibutuhkan sistem *helpdesk ticketing system* yang terintegrasi dengan baik dan cepat sehingga pengaksesan data pada *helpdesk* dapat dilakukan dengan mudah dan cepat guna pengukuran tingkat masalah pengaksesan laporan oleh manajer operasional, serta permasalahan dapat tertangani dengan baik dalam cakupan batasan masalah yang menghasilkan solusi tepat untuk mengatur sumberdaya yang ada. Dalam pengembangan *helpdesk ticketing system* ini, penulis menggunakan metodologi berorientasi pada objek yaitu *iteration waterfall* dengan dimodelkan menggunakan *Unified Modelling Language* (UML). *Tools* yang digunakan adalah XAMPP 1.7.1 dengan spesifikasi Apache 2.2.11 sebagai *web server*, PHP versi 5.2.9 sebagai bahasa pemrograman dan MySQL versi 5.1.33 sebagai *database*. Dengan diterapkannya sistem ini diharapkan *helpdesk* dapat dilakukan secara terkomputerisasi dengan penggunaan sistem sehingga data-data yang terkait dengan *helpdesk* dapat tersimpan secara terpusat dalam *database* dan juga pembuatan laporan yang secara otomatis dalam sistem.

Kata Kunci: *Helpdesk Ticketing System*, *Iteration*, *Waterfall*, Pemodelan berorientasi objek, *Unified Modelling Language* (UML)

V Bab + xxi Halaman + 118 Halaman + 32 Daftar Pustaka + 39 Gambar + 37 Tabel + 6 Daftar Simbol + 5 Lampiran
Pustaka Acuan: 28 Buku (2001-2009)

KATA PENGANTAR

Alhamdulillah, puji syukur penulis panjatkan kepada Allah SWT atas segala rahmat dan karunia-Nya yang diberikan kepada penulis dalam kehidupan di dunia ini sehingga penulis dapat melaksanakan dan menyelesaikan skripsi dengan lancar. Shalawat serta salam penulis curahkan kepada junjungan, suri tauladan, dan Rasul kita baginda Nabi besar Muhammad SAW yang telah menyampaikan ajaran Islam ke muka bumi ini, dan juga kepada keluarga dan para sahabatnya beserta alim ulama yang senantiasa meneruskan ajaran Islam sebagai penerang jalan kehidupan hingga akhir zaman.

Skripsi ini berjudul “Rancang Bangun *Helpdesk Ticketing System* (Studi Kasus: PT. Primus Indojoya)”, yang disusun sebagai salah satu syarat dalam menyelesaikan prograem Strata Satu (S1) pada Program Studi Sistem Informasi di Universitas Islam Negeri Syarif Hidayatullah Jakarta.

Pada kesempatan ini, penulis ingin menyampaikan ucapan terima kasih kepada pihak-pihak yang telah mendukung atas terselesaikannya laporan ini. Karena tanpa dukungan dari mereka, penulis tidak akan mampu menyelesaikan laporan ini dengan baik. Mereka yang telah mendukung penulis adalah:

1. Bapak Dr. Syopiansyah Jaya Putra, M.Sis, selaku Dekan Fakultas Sains dan Teknologi.
2. Ibu Nur Aeni Hidayah, MMSI, selaku Ketua Program Studi Sistem Informasi sekaligus sebagai dosen pembimbing I yang telah banyak memberikan masukan, bimbingan dan motivasi dalam menyelesaikan skripsi ini.

3. Bapak Zainuddin Bey Fananie, MSc, selaku dosen pembimbing II yang telah banyak membantu dalam menyelesaikan skripsi ini, terima kasih banyak untuk waktu dan pengarahannya selama ini.
4. Seluruh Dosen dan Staf Program Studi Sistem Informasi, Fakultas Sains dan Teknologi, yang telah berjasa kepada penulis selama diperkuliahan.
5. Bapak Alex Kurniawan yang telah mengizinkan penulis untuk penelitian di PT. Primus Indojaya, terima kasih sudah menyempatkan waktunya untuk wawancara.
6. Bapak dan Ibuku tercinta dan tersayang, kakak dan adikku, suami dan anakku tercinta, berkat cinta, kasih sayang, dan do'a dari kalian, penulis tidak akan bisa menjadi seperti sekarang ini.
7. Teman-teman seperjuanganku keluarga besar SIB'04 yang sudah menemani penulis selama kuliah, terutama Afrialdi Syahputra yang sudah membantu dalam pembuatan aplikasi. Ade, Yuni, Widya, Nurjanah, Lia, Pipit, Sigit yang berjuang bersama dalam penyelesaian skripsi. Terima kasih untuk semua nya!

Akhir kata penulis menyadari bahwa dalam penulisan laporan ini masih banyak kesalahan-kesalahan dan kekurangan-kekurangan sehingga kritik membangun sangatlah diperlukan. Semoga skripsi ini dapat bermanfaat baik untuk diri penulis maupun untuk kepentingan universitas.

Jakarta, Juli 2011

Qoyyimah

DAFTAR ISI

Halaman Judul.....	i
Lembar Persetujuan.....	ii
Lembar Pengesahan.....	iii
Lembar Pernyataan.....	iv
Abstrak.....	v
Kata Pengantar.....	vi
Daftar Isi.....	viii
Daftar Gambar.....	xii
Daftar Tabel.....	xiv
Daftar Simbol.....	xvi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	4
1.4 Tujuan.....	4
1.5 Manfaat.....	5
1.6 Metodologi Penelitian.....	5
1.6.1 Metode Pengumpulan Data.....	5
1.6.2 Metode Pengembangan Sistem.....	6
1.7 Sistematika Penulisan.....	7
BAB II LANDASAN TEORI.....	9
2.1 Pengertian Rancang Bangun	9
2.2 Pengertian Sistem.....	9
2.3 Pengertian Data.....	10
2.4 Rancangan Sistem.....	10
2.5 Pengertian <i>Helpdesk</i>	11
2.6 Kelebihan-kelebihan <i>Helpdesk</i>	12
2.7 Fungsi <i>Helpdesk Ticketing System</i>	12
2.8 Pengertian <i>Ticketing</i>	13
2.9 UML (<i>Unified Modelling Language</i>).....	13

2.9.1 <i>Usecase Model Diagram</i>	14
2.9.2 <i>Activity Diagram</i>	15
2.9.3 <i>Sequence Diagram</i>	16
2.9.4 <i>Class Diagram</i>	16
2.9.5 <i>Statechart Diagram</i>	16
2.9.6 <i>Deployment Diagram</i>	17
2.10 Pengertian Internet.....	17
2.10.1 Latar Belakang Internet.....	18
2.10.2 Manfaat Internet.....	18
2.11 <i>Website</i>	19
2.11.1 Prinsip-prinsip Desain <i>Website</i>	20
2.11.2 <i>Home Page</i>	23
2.11.3 <i>Web Browser</i>	23
2.11.4 <i>Web Server</i>	24
2.11.5 Dasar-dasar Pemrograman Berbasis <i>Web</i>	24
2.11.6 Keunggulan <i>Web</i> Berfasilitas Basis Data.....	24
2.12 Basis Data (<i>DataBase</i>).....	25
2.12.1 Konsep DBMS (<i>Database Management System</i>).....	26
2.13 Perangkat Lunak Pengembangan Sistem.....	27
2.13.1 <i>Hypertext Preprocessor</i> (PHP)	27
2.13.2 <i>My Structure Query Language</i> (MySQL)	28
2.14 Metodologi Pengumpulan Data	29
2.14.1 Studi Pustaka.....	29
2.14.2 Studi Lapangan.....	29
2.14.2.1 Pengamatan (<i>Observasi</i>).....	29
2.14.2.2 Wawancara (<i>Interview</i>).....	29
2.14.3 Studi Literatur Sejenis.....	30
2.15 Pengembangan Sistem.....	30
2.15.1 Definisi Pengembangan Sistem.....	30
2.15.2 Tahapan-tahapan Sistem.....	30
2.16 Literatur Sejenis.....	33

BAB III METODOLOGI PENELITIAN.....	37
3.1 Metode Pengumpulan Data.....	37
3.1.1 Studi Pustaka.....	37
3.1.2 Studi Lapangan.....	37
3.1.3 Studi Literatur Sejenis.....	38
3.2 Metodologi Pengembangan Sistem.....	39
3.2.1 Permulaan Sistem (<i>System Initiation</i>).....	39
3.2.2 Analisis Sistem (<i>System Analysis</i>).....	41
3.2.3 Desain Sistem (<i>System Design</i>).....	43
3.2.4 Implementasi Sistem (<i>System Implementation</i>).....	44
3.3 Kerangka Berfikir Penelitian.....	45
BAB IV HASIL DAN PEMBAHASAN.....	47
4.1 <i>System Initiation</i> (Permulaan Sistem).....	47
4.1.1 Identifikasi Masalah.....	47
4.1.2 Penetapan Lingkup Sistem.....	48
4.1.3 Penetapan Tujuan Pengembangan Sistem.....	48
4.2 <i>System Analysis</i> (Analisis Sistem).....	49
4.2.1 Gambaran Umum PT. Primus Indojaya	49
4.2.2 Struktur Organisasi.....	50
4.2.3 Analisis Sistem yang Berjalan.....	50
4.2.4 Analisis Kelebihan dan Kekurangan Sistem yang Berjalan.....	52
4.2.4.1 Analisa Perbandingan Sistem.....	52
4.2.4.2 Sistem Literatur Sejenis.....	53
4.2.5 Analisis Usulan Sistem Baru.....	54
4.2.6 Analisis Kebaikan Sistem yang Diusulkan.....	55
4.2.7 Analisis dan Pemodelan Sistem Berorientasi Obyek.....	56
4.2.7.1 Pemodelan <i>Usecase</i> Diagram.....	56
4.2.7.2 Narasi <i>Usecase</i> Analysis.....	59
4.2.7.3 Pemodelan <i>Activity Diagram</i>	66
4.3 <i>System Design</i> (Sistem Desain).....	72
4.3.1 Desain dan Pemodelan Berorientasi Obyek.....	72
4.3.1.1 Narasi <i>Usecase Design</i>	72

4.3.1.2	Pemodelan <i>Class Diagram</i>	80
4.3.1.3	Pemodelan <i>Sequence Diagram</i>	81
4.3.1.4	Pemodelan <i>Statechart Diagram</i>	90
4.3.1.5	Pemodelan <i>Deployment Diagram</i>	94
4.3.2	Desain <i>Database</i> Sistem.....	94
4.3.2.1	<i>Physical Data Model</i> (PDM).....	94
4.3.2.2	<i>Data Definition Language</i> (DDL).....	97
4.3.3	Desain Antarmuka Sistem.....	108
4.4	<i>System Implementation</i> (Implementasi Sistem).....	109
4.4.1	Kontruksi Perangkat Lunak.....	109
4.4.2	Pengujian Perangkat Lunak.....	110
4.4.2.1	<i>Alpha Testing</i>	110
4.4.2.2	<i>Beta Testing</i>	111
4.4.3	Implementasi Perangkat Lunak.....	111
4.4.3.1	Penyiapan Rencana Implementasi Jaringan.....	111
4.4.3.2	Spesifikasi <i>Hardware</i> dan Kebutuhan <i>Software</i>	112
BAB V	PENUTUP.....	114
5.1	Kesimpulan.....	114
5.2	Saran.....	115
	DAFTAR PUSTAKA.....	116
	LAMPIRAN	

DAFTAR GAMBAR

Gambar 3.1 Kerangka Berpikir Penelitian.....	46
Gambar 4.1 Struktur Organisasi PT. Primus Indojoya.....	50
Gambar 4.2 <i>Standart Operation Procedure</i> (SOP) yang berjalan.....	51
Gambar 4.3 <i>Standart Operation Procedure</i> (SOP) Sistem Usulan.....	54
Gambar 4.4 <i>Usecase Model Diagram Helpdesk Ticketing System</i>	60
Gambar 4.5 <i>Activity Diagram Customer</i>	68
Gambar 4.6 <i>Activity Diagram Login</i>	69
Gambar 4.7 <i>Activity Diagram Input Ticket</i>	70
Gambar 4.8 <i>Activity Diagram View Ticket</i>	71
Gambar 4.9 <i>Activity Diagram Assign Ticket</i>	72
Gambar 4.10 <i>Activity Diagram Input Solusi</i>	73
Gambar 4.11 <i>Activity Diagram View Laporan</i>	74
Gambar 4.12 <i>Class Diagram Helpdesk Ticketing System</i>	83
Gambar 4.13 <i>Sequence Diagram Register Helpdesk Ticketing System</i>	84
Gambar 4.14 <i>Sequence Diagram Login Client Helpdesk Ticketing System</i>	84
Gambar 4.15 <i>Sequence Diagram Login Frontdesk Helpdesk Ticketing System</i>	84
Gambar 4.16 <i>Sequence Diagram Login Teknisi Helpdesk Ticketing System</i>	85
Gambar 4.17 <i>Sequence Diagram Login Manager Helpdesk Ticketing System</i>	85
Gambar 4.18 <i>Sequence Diagram Input Ticket Helpdesk Ticketing System</i>	86
Gambar 4.19 <i>Sequence Diagram View Ticket Customer Helpdesk Ticketing System</i>	86
Gambar 4.20 <i>Sequence Diagram View Ticket Frontdesk Helpdesk Ticketing System</i>	87
Gambar 4.21 <i>Sequence Diagram View Ticket Teknisi Helpdesk Ticketing System</i>	87
Gambar 4.22 <i>Sequence Diagram View Ticket Manager Helpdesk Ticketing System</i>	88
Gambar 4.23 <i>Sequence Diagram Assigned Ticket Customer Helpdesk Ticketing System</i>	88

Gambar 4.24 <i>Sequence Diagram Assigned Ticket Frondesk Helpdesk Ticketing System</i>	89
Gambar 4.25 <i>Sequence Diagram Assigned Ticket Teknisi Helpdesk Ticketing System</i>	89
Gambar 4.26 <i>Sequence Diagram Assigned Ticket Manager Helpdesk Ticketing System</i>	90
Gambar 4.27 <i>Sequence Diagram Input Solusi Customer Helpdesk Ticketing System</i>	90
Gambar 4.28 <i>Sequence Diagram Input Solusi Teknisi Helpdesk Ticketing System</i>	91
Gambar 4.29 <i>Sequence Diagram Input Solusi Manager Helpdesk Ticketing System</i>	91
Gambar 4.30 <i>Sequence Diagram Lihat Laporan Helpdesk Ticketing System</i>	92
Gambar 4.31 <i>Statechart Diagram Register Client</i>	92
Gambar 4.32 <i>Statechart Diagram Login</i>	93
Gambar 4.33 <i>Statechart Diagram Ticket</i>	94
Gambar 4.34 <i>Statechart Diagram User</i>	94
Gambar 4.35 <i>Statechart Diagram Laporan</i>	95
Gambar 4.36 <i>Deployment Diagram Helpdesk Ticketing System</i>	96
Gambar 4.37 <i>Physical Data Model Helpdesk Ticketing System</i>	98
Gambar 4.38 <i>Gambaran Implementasi Jaringan Untuk Helpdesk Ticketing System</i>	114

DAFTAR TABEL




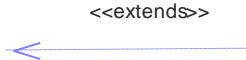
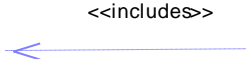
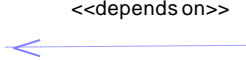
Tabel 4.1 Analisa Perbandingan Sistem.....	53
Tabel 4.2 Perbandingan Literatur Sejenis.....	53
Tabel 4.3 Daftar Pelaku (<i>Actor</i>) Pada <i>Helpdesk Ticketing System</i>	55
Tabel 4.4 Daftar <i>Use Case Helpdesk Ticketing System</i>	56
Tabel 4.5 Daftar Istilah <i>Usecase Helpdesk Ticketing System</i>	59
Tabel 4.6 Narasi <i>Usecase Analysis Register</i>	61
Tabel 4.7 Narasi <i>Usecase Analysis Login</i>	62
Tabel 4.8 Narasi <i>Usecase Analysis Input Ticket</i>	63
Tabel 4.9 Narasi <i>Usecase Analysis View Ticket</i>	64
Tabel 4.10 Narasi <i>Usecase Analysis Assigned Ticket</i>	64
Tabel 4.11 Narasi <i>Usecase Analysis Input Solusi</i>	65
Tabel 4.12 Narasi <i>Usecase Analysis Lihat Laporan</i>	66
Tabel 4.13 Narasi <i>Usecase Design Register</i>	75
Tabel 4.14 Narasi <i>Usecase Design Login</i>	76
Tabel 4.15 Narasi <i>Usecase Design Input Ticket</i>	77
Tabel 4.16 Narasi <i>Usecase Design View Ticket</i>	78
Tabel 4.17 Narasi <i>Usecase Design Assign Ticket</i>	79
Tabel 4.18 Narasi <i>Usecase Design Input Solusi</i>	80
Tabel 4.19 Narasi <i>Usecase Design Lihat Laporan</i>	81
Tabel 4.20 Struktur Tabel <i>Gender</i>	100
Tabel 4.21 Struktur Tabel <i>Industry</i>	100
Tabel 4.22 Struktur Tabel <i>Language</i>	100
Tabel 4.23 Struktur Tabel <i>Level Support</i>	101
Tabel 4.24 Struktur Tabel <i>Priority</i>	101
Tabel 4.25 Struktur Tabel <i>Status</i>	102
Tabel 4.26 Struktur Tabel <i>Role</i>	102
Tabel 4.27 Struktur Tabel <i>User</i>	102
Tabel 4.28 Struktur Tabel <i>Client</i>	103
Tabel 4.29 Struktur Tabel <i>Departement</i>	104
Tabel 4.30 Struktur Tabel <i>Organization</i>	104

Tabel 4.31 Struktur Tabel <i>Ticket</i>	104
Tabel 4.32 Struktur Tabel <i>Message</i>	105
Tabel 4.33 Struktur Tabel <i>Response</i>	105
Tabel 4.34 Struktur Tabel <i>Ticket Status</i>	106
Tabel 4.35 Matriks <i>Data-to-Location-CRUD</i>	106
Tabel 4.36 Daftar <i>Tools</i> Pengembangan Perangkat Lunak Sistem.....	111
Tabel 4.37 Spesifikasi <i>Hardware</i> untuk Implementasi Perangkat Lunak Sistem.....	114




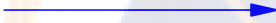




DAFTAR SIMBOL

SIMBOL *USE-CASE MODEL DIAGRAMS*

Simbol	Nama Simbol	Keterangan
 Actor	<i>Actor</i>	Segala sesuatu yang perlu berinteraksi dengan sistem untuk pertukaran informasi.
 Usecase	<i>Usecase</i>	Urutan langkah-langkah yang secara tindakan saling terkait, baik terotomatisasi maupun manual, untuk tujuan melengkapi satu tugas bisnis tunggal.
	<i>Association</i>	<i>Association</i> adalah hubungan interaksi antara <i>actor</i> dengan <i>usecase</i> .
 <<extends>>	<i>Extends</i>	Hubungan antar <i>usecase</i> yang terjadi akibat perluasan fungsi dari salah satu <i>usecase</i> .
 <<includes>>	<i>Includes</i>	Hubungan antar <i>usecase</i> yang terjadi agar tidak terjadi perulangan penggunaan suatu <i>usecase</i> .
 <<depends on>>	<i>Depends on</i>	Hubungan antar <i>usecase</i> yang menggambarkan ketergantungan suatu <i>usecase</i> dengan <i>usecase</i> lainnya.


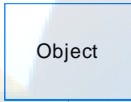


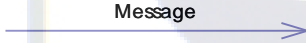


Sumber: Whitten, 2004:258

SIMBOL ACTIVITY DIAGRAM

Simbol	Nama Simbol	Keterangan
	Proses Mulai	Menggambarkan awal sebuah proses.
	Inisiasi Aktivitas	Menggambarkan sasaran yang mengawali kegiatan.
	Aktivitas	Menggambarkan sebuah aktivitas atau tugas yang perlu dilakukan.
	Garis Sinkronisasi	Menggambarkan kegiatan yang dapat muncul secara paralel.
	Aktivitas Keputusan	Menggambarkan sebuah aktivitas keputusan.
	Proses Selesai	Menggambarkan akhir dari sebuah proses.

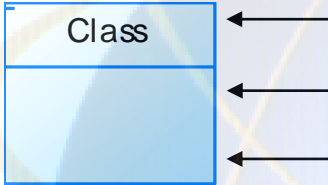

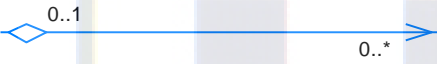

Sumber: Whitten, 2004:428

SIMBOL *SEQUENCE DIAGRAM*

Simbol	Nama Simbol	Keterangan
	<i>Actor</i>	Objek <i>actor</i> yang terdapat dalam sistem.
	<i>Object</i>	Objek (<i>model</i> atau <i>view</i> atau <i>controller</i>) yang terdapat dalam sistem.
	<i>Lifeline</i>	<i>Lifeline</i>
	<i>Behavior (Operation)</i>	Perilaku yang perlu dilakukan oleh masing-masing obyek.
	<i>Message</i>	Pesan yang telah dikirim ke satu obyek tertentu untuk melakukan suatu <i>behavior</i> tertentu.
	<i>Message to Self</i>	Pesan yang dikirimkan pada dari dan kepada obyek itu sendiri.
	<i>Return Message</i>	Pesan yang dikirimkan sebagai balasan dari pesan sebelumnya.



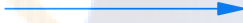
Sumber: Whitten, 2004:664

SIMBOL *CLASS DIAGRAM*

Simbol	Nama Simbol	Keterangan
	<p><i>Class</i></p> <ol style="list-style-type: none"> 1. <i>Class name</i> 2. <i>Attributes</i> 3. <i>Behaviors</i> 	Satu set obyek yang memiliki atribut dan <i>behavior</i> yang sama.
	<i>Association</i>	Hubungan dua arah antar kelas.
	<i>Agregation</i>	Hubungan di mana suatu kelas yang paling besar berisi satu atau lebih kelas yang lebih kecil.
	<i>Generalization/ Specialization</i>	Hubungan yang menggambarkan suatu kelas adalah pewarisan dari kelas lainnya.

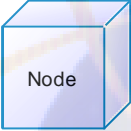
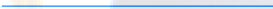
Sumber: Whitten, 2004:411

SIMBOL *STATE* DIAGRAM

Simbol	Nama Simbol	Keterangan
	<i>State Awal</i>	Awal obyek terbentuk.
	Kondisi Obyek	Siklus hidup obyek.
	Alur Perubahan <i>State</i>	Transisi perubahan siklus obyek.
	<i>State Akhir</i>	Keadaan akhir obyek.

Sumber: Whitten, 2004:662

SIMBOL *DEPLOYMENT DIAGRAM*

Simbol	Nama Simbol	Keterangan
	<i>Node</i>	Node dalam sistem.
	<i>Node Association</i>	Menggambarkan bagaimana komunikasi antar <i>node</i> .

Sumber: Whitten, 2004:667

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan yang pesat di bidang teknologi komputer, elektronik, telekomunikasi maupun mekanik telah menghasilkan berbagai aplikasi canggih dan cerdas yang merubah kehidupan manusia pada saat ini dan mendatang (Fajar dan Azman, 2009). Seiring dengan perkembangan jaman, peranan teknologi informasi dan komunikasi sangat berdampak positif dalam meningkatkan kualitas hidup manusia.

Banyak hal yang diusahakan oleh pihak manajemen suatu perusahaan untuk meningkatkan efisiensi. Mulai dari waktu kerja sampai dengan suku cadang mesin dikontrol untuk tujuan efisiensi. Selain mengontrol kerja mesin, sistem monitoring juga bisa dimanfaatkan (Hendrik, 2002). *HelpDesk Ticketing System* sebagai sebuah proses untuk mengumpulkan data dari berbagai sumber yang ada dan *helpdesk* dituntut untuk aktif memonitor dan merawat kebutuhan pengguna. Adapun *helpdesk* bertujuan untuk memudahkan semua penggunaan perangkat IT dalam ruang lingkup perusahaan maupun lembaga instansi pemerintah yang melaporkan setiap permasalahan yang dihadapi. Setiap permasalahan yang telah selesai ditindaklanjuti akan secara otomatis terdokumentasi dan dapat dijadikan referensi serta dapat menyajikan laporan untuk memenuhi kebutuhan informasi dalam suatu perusahaan maupun lembaga instansi pemerintah yang dapat diakses dengan cepat dan mudah. Sehingga, dapat menghasilkan solusi yang tepat dalam mengatur sumber daya yang ada.

Saat ini telah banyak perusahaan-perusahaan besar yang menerapkan aplikasi *report* untuk mengetahui kinerja perusahaan. *Report* tersebut dapat sangat bermanfaat bagi perusahaan dengan cara melaporkan apa saja masalah yang muncul. Jumlah serta kelengkapannya, sehingga perusahaan dapat mencari cara mengantisipasinya suatu waktu. *Report* yang dapat membantu pihak manajemen untuk melihat perkembangan perusahaan.

Akan tetapi, saat ini belum ada satupun aplikasi yang digunakan oleh PT. Primus Indojoya untuk memberikan laporan secara sistematis. Semua pencatatan laporan, laporan bulanan serta perhitungan masalah masih dilakukan secara manual. Hal ini dapat memberi efek yang negatif, yakni: terdapat masalah yang beresiko dicatat secara berulang, adanya masalah yang tidak dicatat karena sibuknya operator dalam menerima telepon, perhitungan masalah tidak akurat karena dilakukan secara *manual* dan perlunya manajerial menunggu untuk mendapatkan laporan yang diinginkan.

Dari permasalahan di atas, maka dibutuhkan *HelpDesk Ticketing System* yang terintegrasi dengan baik sehingga pengaksesan data pada *helpdesk* dapat dilakukan dengan mudah dan cepat guna pengukuran tingkat masalah serta pengaksesan laporan oleh Kepala Bidang IT, serta permasalahan dapat tertangani dengan baik dalam cakupan batasan masalah yang menghasilkan solusi tepat untuk mengatur sumber daya yang ada, dengan aplikasi ini akan jelas masalah-masalah apa yang dihadapi oleh *Client*. Untuk mempermudah agar dapat di akses oleh pihak-pihak yang memerlukannya, aplikasi ini akan di bangun berbasiskan *web* dan dipublikasikan secara umum pada IT *HelpDesk*. Hal ini dapat membantu pihak manajerial dalam mengakses *report* suatu waktu tanpa harus menunggu

pihak IT *HelpDesk* melaporkannya. *HelpDesk Ticketing System* pada PT. Primus Indojaya dapat berfungsi sebagai solusi permasalahan dari *Client* yang mengalami masalah. Kemudian operator akan mencatat *detail* masalah dan akan mencari solusinya. Setiap masalah yang dapat terselesaikan akan diberi “*kode*” dan yang belum akan di beri “*kode*” juga. Karena tidak semua masalah dapat dicari solusinya saat itu juga. Masalah- masalah yang di catat oleh operator di kumpulkan dalam suatu basis data dan secara periodik sistem IT *HelpDesk Ticketing System* akan melaporkan masalah-masalah ini dalam bentuk laporan bulanan kepada pihak manajerial. Diharapkan dengan melihat laporan ini, akan membantu pihak manajerial dalam mengambil keputusan.

Maka penulis mencoba mengungkapkan obyek dari studi dengan judul “**Rancang Bangun *HelpDesk Ticketing System* (Studi Kasus: PT. Primus Indojaya)**”.

1.2 Rumusan Masalah

Berdasarkan penjabaran tersebut maka dapat dirumuskan bahwa masalah yang saat ini dihadapi adalah:

1. Hasil laporan tidak sistematis, sehingga kepada Kepala Bidang IT mengalami kesulitan dalam mengevaluasi hasil laporan.
2. Tidak adanya pengukuran tingkat masalah pada *helpdesk* dalam menangani permasalahan yang ada baik terkait koneksi jaringan maupun kerusakan *hardware* pada PT. Primus Indojaya.
3. Bagaimana cara kerja *Helpdesk Ticketing System* yang baik dan menarik sehingga mempermudah pelanggan untuk mengaksesnya?

1.3 Batasan Masalah

Berdasarkan masalah di atas maka dalam penulisan skripsi ini, penulis membatasi masalah sebagai berikut:

1. Lingkup pembuatan aplikasi ini hanya sebatas pada IT *helpdesk* saja.
2. Pengembangan sistem ini hanya memberikan laporan bulanan kepada Kepala Bidang IT dari permasalahan yang terjadi.
3. Pengukuran tingkat masalah dilihat dari jenis masalah, masalah koneksi jaringan internet dan kerusakan *hardware* yang terkoordinasi.
4. Dalam penelitian ini, penulis menggunakan pemrograman PHP (*Page Hypertext Preprocessor*) dan MySQL sebagai *database*-nya.
5. Penelitian ini tidak membahas keamanan data dan pemeliharaan (*maintenance*) serta pengujian sistem dilakukan dengan *blackbox testing*, dimana penulis melakukan pengujian secara pribadi.
6. Penelitian ini hanya menggunakan metodologi *waterfall* dengan pendekatan UML.

1.4 Tujuan

Tujuan dari pelaksanaan penelitian ini adalah :

1. Untuk membantu pihak IT *HelpDesk* memberikan laporan bulanan.
2. Untuk membantu pihak manajemen mengambil keputusan dengan melihat banyaknya masalah berdasarkan operator tersebut.
3. Untuk mempermudah kerja manajemen dalam mengontrol kinerja operator.

4. Untuk mempermudah kerja operator IT *HelpDesk*, karena dengan aplikasi ini operator tidak perlu lagi melakukan banyak proses dalam memasukkan data.

1.5 Manfaat

Manfaat dari pelaksanaan penelitian ini adalah :

1. Membantu manajer untuk mendapatkan laporan bulanan.
2. Meningkatkan kinerja perusahaan dengan adanya *report* tiap bulan dan dapat diakses kapanpun karena berbentuk web.
3. Mampu mengetahui pengukuran tingkat masalah pada permasalahan yang ditangani oleh *helpdesk*.
4. Perhitungan frekuensi masalah lebih sistematis.
5. Menghindari kesalahan perhitungan karena operasi *manual* pada saat menghitung frekuensi masalah.

1.6 Metodologi Penelitian

Adapun metode yang digunakan dalam penelitian ini meliputi dua bagian, yaitu metode pengumpulan data dan metode pengembangan sistem. Dalam mengumpulkan data untuk pengembangan sistem ini dilakukan dengan cara:

1.6.1 Metode Pengumpulan Data

Merupakan metode yang digunakan peneliti dalam melakukan analisis data dan menjadikannya informasi yang akan digunakan untuk mengetahui permasalahan yang dihadapi.

1. Studi Pustaka

Dilakukan dengan cara membaca buku-buku, mengumpulkan data dari situs *internet*, serta artikel dan penelitian yang berhubungan dengan topik yang akan dibahas dalam penyusunan skripsi ini.

2. Studi Lapangan

- a. Observasi

Pengamatan yang langsung dilakukan oleh peneliti terhadap obyek-obyek yang ada pada perusahaan atau instansi untuk mendapatkan data-data yang diperlukan (Hartono, 2005).

- b. Wawancara

Mengumpulkan data dengan mewawancarai langsung orang yang terkait langsung dengan penggunaan sistem IT *HelpDesk* (Hartono, 2005).

3. Studi Literatur Sejenis

Metode pengumpulan data dengan membandingkan *research* yang sejenis dengan judul penelitian, diantaranya yaitu adalah “Pengembangan Sistem Monitoring *HelpDesk* pada Pustekkom Depdiknas”.

1.6.2 Metode Pengembangan Sistem

Rancang Bangun *HelpDesk Ticketing System* Dengan Pendekatan Client Relationship Management, menggunakan strategi *iteration waterfall* dengan metode analisis dan desain berorientasi obyek (Whitten, 2004), meliputi tahapan-tahapan yang dilakukan secara berulang-ulang hingga sistem telah sesuai dengan kebutuhan pengguna. Tahapan-tahapan tersebut yaitu:

1. Permulaan sistem (*System Initiation*)

Pada tahapan ini penulis melakukan inisiasi terhadap studi kepustakaan, observasi, wawancara, identifikasi yang terjadi serta tujuan pengembangan pada penelitian ini.

2. Analisis sistem (*System Analysis*)

Pada tahap ini penulis memahami sistem yang sedang berjalan, analisis sistem serta dibuat usulan pada sistem baru dengan memberikan solusi.

3. Desain Sistem (*System Design*)

Pada tahapan ini penulis melakukan perancangan proses sebagai alternatif solusi, spesifikasi proses. Kemudian merancang *database* dan tampilan sebagai desain dari pemilihan solusi terbaik, serta perancangan jaringan.

4. Implementasi sistem (*System Implementation*)

Pada tahapan ini penulis melakukan *coding* dari implementasi solusi yang dipilih dan mengevaluasi hasilnya dengan pengujian sistem. Apabila terdapat masalah yang tidak terpecahkan maka penulis melakukan pengkajian ulang ke langkah identifikasi masalah dan analisa sistem.

1.7 Sistematika Penulisan

Dalam penulisan skripsi ini, penulis menjabarkan penelitian Rancang Bangun *HelpDesk Ticketing System* dengan Pendekatan CRM pada PT. Primus Indojaya. dibagi menjadi 5 (lima) Bab, diantaranya adalah:

BAB I PENDAHULUAN

Dalam bab ini akan diuraikan tentang latar belakang, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian serta sistematika penulisan.

BAB II LANDASAN TEORI

Dalam bab ini akan diuraikan mengenai landasan teori yang digunakan dalam pembahasan penulisan skripsi ini dan sumber landasan teori tersebut.

BAB III METODOLOGI PENELITIAN

Bab ini berisi metodologi penelitian yang dilakukan serta langkah-langkah yang digunakan terkait dengan penelitian yang dilakukan.

BAB IV PEMBAHASAN

Dalam bab ini berisi tentang analisis dan perancangan kebutuhan sistem dari hasil penelitian serta pembahasan yang mencakup gambaran umum tentang obyek penelitian serta implementasi sistem.

BAB V PENUTUP

Bab ini merupakan akhir penulisan skripsi, di mana berdasarkan uraian telah dibahas akan dituangkan ke dalam suatu bentuk simpulan akhir serta saran-saran.

BAB II

LANDASAN TEORI

2.1 Pengertian Rancang Bangun

Dalam kamus besar bahasa Indonesia, rancang atau merancang dapat diartikan sebagai mengatur atau merencanakan segala sesuatu (sebelum bertindak, mengerjakan atau melakukan sesuatu), yang akan menghasilkan sebuah rancangan dalam bentuk program aplikasi.

Sedangkan bangun dapat diartikan sebagai cara dalam menyusun atau susunan yang merupakan suatu wujud, struktur, dan sebagainya. Jadi rancang bangun adalah mengatur atau merencanakan segala sesuatu untuk menyusun suatu struktur yang ada untuk menghasilkan sebuah rancangan dalam bentuk program aplikasi (Departemen Pendidikan dan Kebudayaan, 1991).

2.2 Pengertian Sistem

Sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan (McLeod et al, 2001). Sedangkan pengertian sistem menurut Sutabri (2005) adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Sistem adalah kumpulan dari komponen atau elemen yang saling berhubungan satu dengan lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu (Jogiyanto, 2005). Simpulan dari tiga definisi tersebut mengenai sistem adalah sekelompok elemen atau unsur yang saling terkait menjadi satu untuk mencapai suatu tujuan tertentu.

2.3 Pengertian Data

Data adalah bahan baku informasi, didefinisikan sebagai kelompok teratur simbol-simbol yang mewakili kuantitas, tindakan, benda, dan sebagainya yang disusun untuk diolah dalam bentuk struktur data, struktur *file* dan basis data. (Wahono, 2004).

2.4 Rancangan Sistem

Perancangan sistem adalah proses penyiapan spesifikasi yang terperinci untuk mengembangkan sistem yang baru. (Ladjamudin, 2005). Spesifikasi tersebut meliputi:

- a) Spesifikasi keluaran sistem, yang didalamnya mencakup isi, format, *volume*, frekuensi laporan-laporan dan dokumen-dokumen.
- b) Desain semua hal yang penting mengenai langkah-langkah pengolahan, prosedur-prosedur dan pengendalian.
- c) Penyiapan rencana implementasi sistem yang baru.

Perancangan sistem ditujukan untuk menghilangkan kekurangan dan meningkatkan kelebihan sistem yang sedang berjalan. (Jogiyanto, 2005).

Rancangan sistem dapat diartikan sebagai berikut (Jogiyanto, 2005):

- a) Tahap setelah analisis dari siklus pengembangan sistem.
- b) Pendefinisian dari kebutuhan-kebutuhan fungsional.
- c) Persiapan untuk rancang bangun implementasi.
- d) Menggambarkan bagaimana suatu sistem dibentuk.

- e) Yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi.
- f) Termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem.

Dengan demikian maka dapat disimpulkan bahwa perancangan sistem adalah proses penyiapan spesifikasi yang terperinci untuk menghindari kekurangan dan meningkatkan kelebihan sistem yang sedang berjalan atau belum ada sebelumnya dalam membangun sistem baru.

2.5 Pengertian *Helpdesk*

HelpDesk pada dasarnya adalah *Point* dimana masalah atau *issue* dilaporkan dan diatur secara terurut dan diorganisasikan dari perspektif umum. *Helpdesk* merupakan bagian pelengkap dari sebuah fungsi pelayanan dan bertanggung jawab sebagai sumber dari pemecahan masalah atau *issue* lainnya. (<http://www.help-desk-world.com/help-desk.htm>).

HelpDesk merupakan bagian dari perusahaan yang menyediakan dokumen fungsi produk, servis dan teknologi dari perusahaan tersebut.

HelpDesk adalah sebuah departemen atau bagian dalam perusahaan yang melayani atau menanggapi pertanyaan teknis. *HelpDesk* digunakan untuk menjawab pertanyaan-pertanyaan dari cabang, pertanyaan dan jawaban dapat disampaikan melalui telepon dan email.

HelpDesk adalah titik hubungan dalam suatu organisasi dan para pegawai yang menghubungi *helpdesk* dapat menemukan jawaban atas pertanyaan-

pertanyaan mereka dan solusi-solusi dari masalah-masalah yang dihadapi, seperti masalah-masalah di tempat kerja, masalah penggunaan komputer, pekerjaan dengan aplikasi *software*, atau ke sebuah printer dan pertanyaan-pertanyaan teknis lainnya. (<http://www.remedy.com>).

2.6 Kelebihan-kelebihan *HelpDesk*

HelpDesk memiliki beberapa kelebihan, antara lain : (<http://www.help-desk-world.com/help-desk-htm>)

1. *HelpDesk* dapat memberikan solusi atas pertanyaan-pertanyaan maupun keluhan yang masuk dalam waktu yang lebih singkat.
2. *HelpDesk* dapat mengecek status permasalahan yang ada dan mengatur pembagian kerja staf.
3. *HelpDesk* dapat meningkatkan efisiensi perusahaan dalam menangani pertanyaan dan keluhan dari cabang.
4. *HelpDesk* memberi laporan kerja perkembangan kinerja para staf kepada pemimpin.

2.7 Fungsi *Helpdesk Ticketing System*

Helpdesk Ticketing System memiliki beberapa fungsi, antara lain:

1. Dari prespektif keuangan adalah untuk meningkatkan keuntungan, meningkatkan jumlah pelanggan, meningkatkan kepuasan pelanggan, meningkatkan mutu pelayanan.
2. Dari perspektif pelanggan adalah untuk menjalin kemitraan dengan pelanggan, meningkatkan kepercayaan pelanggan.

2.8 Pengertian *Ticketing*

Ticketing adalah Sebuah karcis gangguan (kadang-kadang disebut laporan masalah) adalah mekanisme yang digunakan dalam suatu organisasi untuk melacak deteksi, pelaporan, dan resolusi dari beberapa jenis masalah. *Trouble* sistem *ticketing* berasal dari manufaktur sebagai dasar sistem pelaporan kertas, sekarang kebanyakan berbasis *web* dan terkait dengan hubungan manajemen pelanggan (CRM) lingkungan, seperti *call center* atau *e-business* (<http://SearchCRM.techtarget.com/definition/trouble-ticket>).

2.9 UML (*Unified Modelling Language*)

Pada bagian sebelumnya, telah dibahas tentang pengembangan *model-driven* yang di dalamnya terdapat pemodelan berorientasi obyek. Teknik pemodelan obyek menyajikan penggunaan metode dan notasi diagram yang sama sekali berbeda dengan teknik lainnya. UML merupakan pemodelan standar berorientasi obyek yang telah dikembangkan oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson (Whitten, 2004).

Menurut Jeffrey L. Whitten (2004) UML merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan obyek.

UML menawarkan diagram yang dikelompokkan menjadi beberapa perspektif berbeda untuk memodelkan suatu sistem; seperti satu set cetak biru (*blueprint*) yang digunakan untuk membangun sebuah rumah (Whitten, 2004).

Beberapa diagram UML yang digunakan dalam perancangan *helpdesk ticketing system* ini yaitu: *usecase diagram*, *activity diagram*; *class diagram*, *sequence diagram*, *state diagram*, dan *deployment diagram*.

2.9.1 *Usecase Model Diagram*

Usecase adalah teknik untuk merekam persyaratan fungsional sebuah sistem. *Usecase* mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan (Fowler, 2004).

Usecase diagram merupakan diagram yang menggambarkan interaksi antara sistem dengan sistem eksternal dan pengguna. Dengan kata lain, secara grafis menggambarkan siapa yang akan menggunakan sistem dan dengan cara apa pengguna mengharapkan untuk berinteraksi dengan sistem (Whitten, 2004).

Simbol-simbol yang digunakan dalam *usecase diagram* adalah sebagai berikut:

1. *Usecase*

Pemodelan *usecase* mengidentifikasi dan menggambarkan fungsi-fungsi sistem dari sudut pandang pengguna eksternal dan dalam sebuah cara dan terminologi yang mereka pahami. *Usecase* merupakan urutan langkah-langkah yang secara tindakan saling terkait (*scenario*), baik otomatis maupun secara manual (Whitten, 2004).

2. *Actor* (Pelaku)

Actor merupakan segala sesuatu yang perlu berinteraksi dengan sistem untuk pertukaran informasi. *Actor* dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun (Whitten, 2004).

3. *Relationship* (Hubungan)

Pada diagram *usecase*, *relationship* digambarkan sebagai sebuah garis antara dua simbol. Pemaknaan *relationship* berbeda-beda tergantung bagaimana garis tersebut digambar dan tipe simbol apa yang digunakan untuk menghubungkan garis tersebut. Berikut adalah perbedaan diantara *relationship* yang ada pada sebuah diagram *usecase* (Whitten, 2004):

a. *Extends*

Extension usecase merupakan *usecase* yang terdiri dari langkah yang terekstraksi dari *usecase* yang lebih kompleks untuk menyederhanakan masalah dan karena itu memperluas fungsinya.

b. *Includes*

Hubungan *include* menggambarkan bahwa satu *usecase* seluruhnya meliputi fungsionalitas dari *usecase* lainnya.

c. *Depends on*

Hubungan yang memperlihatkan *usecase* mana yang memiliki ketergantungan pada *usecase* lain untuk menetapkan rangkaian *usecase* yang perlu dikembangkan.

2.9.2 Activity Diagram

Activity diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja (Fowler, 2004). *Activity diagram* secara grafis

digunakan untuk menggambarkan rangkaian aliran aktifitas baik proses bisnis atau *usecase* (Whitten, 2004).

Diagram ini berbeda dengan *flowchart* dimana diagram ini menyediakan sebuah mekanisme untuk menggambarkan kegiatan yang tampak secara paralel (Munawar, 2005).

2.9.3 Sequence Diagram

Sequence diagram secara grafis menggambarkan bagaimana *object* berinteraksi dengan satu sama lain melalui pesan pada eksekusi sebuah *usecase* atau operasi. Sebuah *sequence diagram*, secara khusus, menjabarkan *behaviour* (perilaku) sebuah skenario tunggal (Flower, 2004). Diagram ini mengilustrasikan bagaimana pesan terkirim dan diterima di antara *object* dan *sequence* (ruang waktu) (Whitten, 2004).

2.9.4 Class Diagram

Class diagram mendeskripsikan jenis-jenis obyek dalam sistem dan berbagai macam hubungan statis yang terdapat di antara mereka (Fowler, 2004). *Class diagram* merupakan gambar grafis mengenai struktur obyek statis dari suatu sistem, menunjukkan kelas-kelas obyek yang menyusun sebuah sistem dan juga hubungan antara kelas obyek tersebut (Whitten, 2004).

2.9.5 Statechart Diagram

Statechart diagram adalah teknik yang umum digunakan untuk menggambarkan *behavior* sebuah sistem (Fowler, 2004). *State diagram*

mengilustrasikan siklus hidup *object* dan keadaan yang dapat diasumsikan oleh *object* dan *events* yang menyebabkan *object* beralih dari satu *state* ke *state* yang lain (Whitten, 2004).

Tidak semua obyek yang terdapat pada sistem dibuat *state diagram*-nya, hanya obyek yang memiliki perubahan status yang akan dibuat *state*-nya dan *state diagram* digunakan hanya untuk dokumentasi (Bogs, 2002).

2.9.6 Deployment Diagram

Deployment diagram menunjukkan susunan fisik sebuah sistem (Fowler, 2004). *Deployment diagram* digunakan untuk memahami susunan fisik sistem dan mengkomunikasikan bagaimana sistem akan dikembangkan kepada pengguna (Bogs, 2002).

Seluruh simbol yang terdapat pada diagram-diagram di atas dapat dilihat pada halaman daftar simbol.

2.10 Pengertian Internet

Internet merupakan dua komputer atau lebih yang saling berhubungan membentuk jaringan komputer hingga meliputi jutaan komputer di dunia (Internasional), yang saling berinteraksi dan bertukar informasi, sedangkan dari segi ilmu pengetahuan, internet merupakan sebuah perpustakaan besar yang di dalamnya terdapat jutaan bahkan milyaran informasi atau data yang dapat berupa teks, grafik, audio, maupun animasi dan lain-lain dalam bentuk media elektronik. Orang dapat “berkunjung” ke perpustakaan tersebut kapan saja dan dimana saja. Dari segi komunikasi, internet adalah sarana yang sangat efisien dan efektif untuk

melakukan pertukaran informasi jarak jauh maupun di dalam lingkungan perkantoran. (Nugroho, 2004).

Awalnya internet dikenal sebagai suatu wadah bagi para peneliti untuk saling bertukar informasi yang kemudian dimanfaatkan oleh perusahaan-perusahaan komersil sebagai sarana bisnis mereka, saat ini pengguna internet tersebar di seluruh dunia dengan jumlah mencapai lebih dari 250 juta orang.

2.10.1 Latar Belakang Internet

Internet pertama kali dikembangkan pada tahun 1969 oleh Departemen Pertahanan Amerika Serikat dalam proyek ARPAnet (*Advance Research Projects Agency*). ARPAnet merupakan sebuah jaringan eksperimen milik pemerintah Amerika Serikat berbasis komunikasi data paket. Tujuan dari proyek ini yaitu untuk menghubungkan para periset ke pusat-pusat komputer, sehingga mereka bisa bersama-sama memanfaatkan sarana komputer seperti *disk space*, *database* dan lain-lain. (Nugroho, 2004).

2.10.2 Manfaat Internet

Secara umum banyak sekali manfaat atau kegunaan jika kita mengakses internet. Dengan adanya internet kita bisa mencari informasi apa saja yang ingin kita cari, berikut ini sebagian dari apa yang ada di internet. Informasi untuk kehidupan pribadi : kesehatan, rekreasi, hobby, pengembangan pribadi, rohani, sosial dan lainnya. Informasi untuk kehidupan professional / kerja : sains, teknologi, perdagangan, saham, komoditas, berita bisnis, asosiasi profesi, asosiasi bisnis dan berbagai forum komunikasi. (Nugroho, 2004).

Manfaat internet terutama diperoleh melalui kerjasama antar pribadi atau kelompok tanpa mengenal batas jarak dan waktu. Untuk lebih meningkatkan kualitas sumber daya manusia, sudah waktunya para profesional memanfaatkan jaringan internet dan menjadi bagian dari masyarakat informasi dunia.

2.11 Website

Website merupakan fasilitas hiperteks untuk menampilkan data berupa teks, gambar, suara, animasi, dan data multimedia lainnya. Penggolongan *website* berdasarkan isinya (*website contents*) terdiri dari dua jenis yaitu (Wahana, 2006):

1. Web Statis

Web statis adalah *web* yang berisi/menampilkan informasi-informasi yang sifatnya statis (tetap). Disebut statis karena pengguna tidak dapat berinteraksi dengan *web* tersebut. Singkatnya, untuk mengetahui suatu *web* lain dan berisi suatu informasi yang tetap maka *web* tersebut disebut statis. Pada *web* statis, pengguna hanya dapat melihat isi dokumen pada halaman *web* dan apabila diklik akan berpindah ke halaman *web* yang lain. Interaksi pengguna hanya terbatas dapat melihat informasi yang ditampilkan, tetapi tidak dapat mengolah informasi yang dihasilkan. *Web* statis biasanya berupa *HyperText Markup Language* (HTML) yang ditulis pada editor teks dan disimpan dalam bentuk .html atau .htm. Pengertian HTML adalah bahasa yang menggunakan perintah sederhana dalam standar dokumen untuk menyediakan suatu tampilan visual yang terintegrasi. HTML terdiri atas perintah-perintah sederhana yang menjelaskan bagaimana struktur dokumen, tetapi tidak memformatnya. *Browser* yang menampilkan HTML

akan memformat dan menyesuaikan tampilan HTML sehingga sesuai dengan layar computer pengunjung. (Wahana, 2006).

2. *Web* Dinamis

Web dinamis adalah *web* yang menampilkan informasi serta dapat berinteraksi dengan pengguna. *Web* yang dinamis memungkinkan pengguna untuk berinteraksi menggunakan form sehingga dapat mengolah informasi yang ditampilkan. *Web* dinamis bersifat interaktif, tidak kaku, dan terlihat lebih indah. *Web* dinamis biasanya berupa *Page Hypertext Preprocessor* (PHP) yang membuat halaman *web* HTML menjadi dinamis. HTML yang digabung dengan *script* PHP akan menghasilkan tampilan *web* yang dinamis, indah, dan interaktif. (Wahana, 2006).

2.11.1 Prinsip-prinsip Desain Website

Ada tujuh prinsip yang harus diperhatikan dalam mendesain sebuah *website* agar memperoleh hasil yang baik dan efektif. (Wahana, 2006) yaitu:

1. *Website* dibuat untuk pengguna

Dalam proses merancang *web* harus memfokuskan desainnya pada kepentingan pengguna (*user*). Hal ini juga berarti, perancang *web* harus menganggap pengguna adalah seorang yang awam terhadap segala aspek yang ada pada *website*. Apa yang diinginkan oleh perancang belum tentu menjadi apa yang diinginkan pengguna. Perancang *web* juga harus mempertimbangkan karakter pengguna yang berbeda satu sama lain. Pengguna yang mengunjungi *website* berasal dari latar belakang,

kebudayaan, pendidikan, dan kepentingan yang berbeda-beda maka desain *web* yang dibuat setidaknya harus mewakili selera sebagian besar pengguna.

2. *Utility dan Usability*

Utility adalah kegunaan atau fungsionalitas suatu *web*. Sedangkan *usability* adalah sifat *website* yang mendukung kemampuan pengguna dalam memanipulasi *website* sehingga pengguna memperoleh apa yang diperlukannya.

Beberapa ciri dari *usability* antara lain: dapat dipelajari dengan mudah, penggunaannya efisien, mudah diingat, dan membuat pengguna menjadi puas.

3. *Correctness*

Correctness maksudnya tidak ada kesalahan dalam penulisan *script website*, antara lain:

- a) Dalam penulisan *script* HTML tidak ada kesalahan.
- b) Gambar-gambar yang ditampilkan sesuai yang diharapkan.
- c) Tidak ada kesalahan dalam sistem navigasi.

4. Batasan media internet dan *web*

a. *Browser*

- 1) Membuat *website* yang biasa dipakai di setiap *browser* dan memperhatikan pilihan setting *browser* para pengguna, sehingga halaman tersebut dapat tampil dengan lebih maksimal.
- 2) Sedapat mungkin buatlah *website* dengan teknologi yang paling *compatible* dan dapat di-load oleh sebagian besar *browser* sehingga bisa diterima oleh lebih banyak pengguna.

b. *Bandwidth*

Untuk menghasilkan *website* yang menarik namun tetap cepat di-*load* maka ini berpengaruh juga terhadap besarnya file total *website* yang akan dibuat, juga termasuk gambar-gambar yang digunakan.

Untuk *website* yang baik, *loading* halaman *web* tidak lebih dari 8 detik dan ukuran file setiap halaman HTML beserta gambar-gambarnya tidak lebih dari 65 Kb, agar pengguna yang mengunjungi situs tersebut tidak menjadi jenuh dan pindah ke situs lain.

5. *Website* harus memperhatikan aspek *Graphical User Interface* (GUI).

Website yang baik harus mudah dipahami pada saat pertama kali pengguna mengunjungi *website* tersebut karena pada saat itulah yang menentukan apakah pengguna akan mengunjungi *website* itu lagi.

6. Struktur *Link* dan Navigasi

Website harus memiliki navigasi dan *link* yang jelas, agar pengguna tidak tersesat karena *link* yang disediakan kurang jelas. Pengguna harus dapat menjelajahi semua halaman dengan mudah, mendapatkan informasi tentang halaman yang sedang dikunjungi dan yang sudah dikunjungi, dan juga perlu diperhatikan kecepatan pengguna dalam mendapatkan informasi yang diinginkan, misalnya dengan cara mengatur *link* sedemikian rupa sehingga pengguna mendapatkan informasi kurang dari 5 kali melakukan klik. Hal ini sangat penting artinya untuk kepuasan dan kenyamanan pengguna terhadap *website*.

7. Alat bantu

a. Peta Situs

Penggunaan peta situs dapat mempermudah pengguna dalam memahami tentang isi dan halaman pada *website*.

b. *Search engine*

Apabila jumlah halaman web mencapai lebih dari 100 halaman, sebaiknya dibuat *search engine* untuk mempermudah pengguna dalam mencari informasi yang diinginkan sehingga pengguna cukup mengetikkan kata kunci untuk informasi yang diinginkan dan informasi dapat segera ditampilkan.

2.11.2 Home Page

Home page adalah halaman utama pada sebuah *website*. Halaman inilah yang akan ditampilkan ketika Anda memanggil sebuah *website* walaupun Anda tidak menyebutkan halaman tersebut. (Kadir, 2003). Dari definisi *home page* di atas, dapat disimpulkan bahwa *home page* merupakan halaman awal atau halaman muka saat pengguna mengunjungi *website* tersebut.

2.11.3 Web Browser

Web browser adalah salah satu jenis program *client* yang dapat mengakses beberapa layanan internet. Jenis *browser* pada saat ini antara lain: *Microsoft Internet Explorer*, *Mozilla Firefox*, *Netscape Navigator*, *Opera* dan *Safari*. (Sembiring, 2001).

2.11.4 Web Server

Web server adalah sebuah bentuk *server* yang khusus digunakan untuk menyimpan halaman *website* atau *homepage*. (Nugroho, 2004). Jenis *web server* yang ada pada saat ini, yaitu: *Apache (open source)* yang dapat berjalan pada sistem operasi Windows dan Linux serta *Internet Information Service (IIS)* yang berjalan pada program Windows.

2.11.5 Dasar-dasar Pemrograman Berbasis Web

Beberapa dasar *web* yang perlu diketahui antara lain (Hariyanto, 2004):

1. Komunikasi antara *web browser* dan *web server* berdasarkan protokol HTTP.
2. Dokumen (bahkan sumber daya apapun di jaringan) yang dikehendaki diidentifikasi dengan URL (*Universal Resource Locator*, masih banyak yang menyebut *Uniform Resource Locator*).
3. Dokumen *web* ditulis berdasarkan standar HTML.
4. Pemrograman sisi *client* (*client-side scripting*) dan *java applet*.
5. Pemrograman sisi *server* (*server-side scripting / programming*).

2.11.6 Keunggulan Web Berfasilitas Basis data

Dengan semakin berkembangnya layanan informasi dan *e-commerce* pada *web*, maka basis data yang digunakan, sistem pendukung keputusan dan pengolahan transaksi harus ditautkan dengan *web*. Formulir HTML merupakan antar muka nyaman untuk pengolahan transaksi. Pemakai dapat mengisi rincian-rincian formulir dan melakukan *klik submit* untuk mengirim pesan ke *server*.

Server mengeksekusi transaksi basis data di *situs server*. *Server* melakukan format hasil menjadi dokumen HTML dan mengirim balik ke pemakai.

Menghubungkan basis data dan *web* penting karena dokumen statik di *situs web* mempunyai keterbatasan bahkan untuk pemakai yang tidak melakukan *query* atau pengolahan transaksi sekalipun (Hariyanto, 2004).

2.12 Basis Data (*DataBase*)

Basis data (*database*) adalah koleksi terpadu dari data-data yang saling berkaitan dari suatu *enterprise* (perusahaan, instansi pemerintah atau swasta). (Ladjamudin, 2005).

Basis data (*database*) terdiri dari data yang akan digunakan atau diperuntukkan terhadap banyak "*user*", dimana masing-masing "*user*" akan menggunakan data tersebut sesuai dengan tugas dan fungsinya, dan "*user*" lain dapat juga menggunakan data tersebut dalam waktu yang bersamaan. (Ladjamudin, 2005).

Basis data adalah suatu penyusunan data terstruktur yang disimpan dalam media pengingat (*hardisk*) yang tujuannya adalah agar data tersebut dapat di akses dengan mudah dan cepat. (Kadir, 2003).

Beberapa keuntungan dari basis data terhadap sistem pemrosesan berkas. (Ladjamudin, 2005) :

1. Kemubaziran data berkurang.
2. Penggunaan data lebih mudah.
3. Sekuriti data lebih mudah dilakukan.
4. Berbagi data dapat selalu dilakukan oleh setiap "*user*".

Beberapa kelemahan dari basis data (<http://kamii-yogyakarta.tripod.com/tools.htm>) adalah sebagai berikut:

3. *Storage* yang digunakan menjadi besar.
4. Dibutuhkan tenaga yang terampil dalam mengelola data.
5. Perangkat lunaknya mahal.
6. Kerusakan pada sistem basis data dapat mempengaruhi departemen yang terkait.
7. Terjadi *deadlock*.

Beberapa Tujuan Basis data (<http://kamii-yogyakarta.tripod.com/tools.htm>) adalah sebagai berikut:adalah:

1. Efisiensi meliputi *speed*, *speace* dan *accuracy*.
2. Menangani data dalam jumlah besar.
3. Kebersamaan pemakai (*Sharebility*).
4. Meniadakan duplikasi dan data yang tidak konsisten.

2.12.1 Konsep DBMS (*Database Management System*)

Kumpulan atau gabungan *database* dengan perangkat lunak dinamakan *Database Management System* atau disingkat menjadi DBMS. DBMS merupakan koleksi terpadu dari *database* dan program-program komputer yang digunakan untuk mengakses dan memelihara *database*. Program-program tersebut menyediakan berbagai fasilitas operasi untuk memasukkan, melacak, dan memodifikasi data kedalam *database*, mendefinisikan data baru, serta mengolah data menjadi informasi yang dibutuhkan. (Ladjamudin, 2005).

Tujuan utama dari DBMS adalah untuk menyediakan suatu lingkungan yang mudah dan efisien untuk penggunaan, penarikan, dan penyimpanan data dan informasi. (Ladjamudin, 2005).

Dibandingkan dengan sistem tradisional yang berbasis kertas, DBMS memiliki empat keunggulan sebagai berikut (Lajamudin, 2005) :

1. Kepraktisan, sistem yang berbasis kertas akan menggunakan kertas yang sangat banyak untuk menyimpan informasi, sedangkan DBMS menggunakan media penyimpanan sekunder yang berukuran kecil tetapi pada informasi.
2. Kecepatan, mesin dapat mengambil atau mengubah data jauh lebih cepat dari pada manusia.
3. Mengurangi kejemuian, orang cenderung menjadi bosan kalau melakukan tindakan-tindakan yang berulang yang menggunakan tangan (misalnya harus mengganti suatu informasi).
4. Kekinian, informasi yang tersedia pada DBMS akan bersifat mutakhir dan akurat setiap saat.

2.13 Perangkat Lunak Pengembangan Sistem

2.13.1 *Hypertext Preprocessor (PHP)*

PHP merupakan bahasa pemrograman berbentuk script yang ditempatkan dalam server dan diproses di server. Hasil dari pengolahan akan dikirimkan ke klien, tempat pemakai menggunakan *browser*. Secara khusus, PHP dirancang untuk membentuk *web* dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, kita bisa menampilkan isi *database* ke

halaman web. Pada prinsipnya, PHP mempunyai fungsi yang sama dengan *script* seperti ASP (Active Server Page), Cold Fusion, ataupun Perl (Kadir, 2003).

2.13.2 *My Structure Query Language (MySQL)*

Salah satu aplikasi basis data adalah MySQL yang merupakan sebuah program pembuat basis data yang bersifat *open source*, artinya siapa saja boleh menggunakannya dan tidak dicekal. (Nugroho, 2004). MySQL merupakan sistem manajemen basis data yang fungsinya untuk menambah, mengakses, dan memproses data yang disimpan dalam sebuah basis data komputer. Kelebihan yang dimiliki MySQL yaitu:

- a. Ditulis dalam bahasa C dan C++.
- b. Bekerja dalam berbagai platform (misalnya Windows, Mac OS X, Solaris, Unix, dan lain-lain).
- c. Menyediakan mesin penyimpanan (*engine storage*) transaksi dan non transaksi.
- d. *Server* tersedia sebagai program yang terpisah untuk digunakan pada lingkungan jaringan klien/*server*.
- e. Mempunyai *library* yang dapat ditempelkan pada aplikasi yang berdiri sendiri (*standalone application*) sehingga sistem tersebut dapat digunakan pada komputer yang tidak mempunyai jaringan.
- f. Mempunyai sistem *password* yang fleksibel dan aman.
- g. Klien dapat terkoneksi ke MySQL *server* menggunakan soket TCP/IP pada *platform* mana pun.

- h. *Server* dapat mengirim pesan kesalahan ke klien dalam berbagai bahasa.

MySQL termasuk jenis RDBMS (*Relational Database Management System*). Pada MySQL sebuah basis data terdiri atas tabel-tabel dan sebuah tabel terdiri atas baris dan kolom.

2.14 Metodologi Pengumpulan Data

2.14.1 Studi Pustaka

Studi pustaka adalah segala usaha yang dilakukan oleh peneliti untuk menghimpun informasi yang relevan dengan topik atau masalah yang akan atau sedang diteliti. Informasi itu dapat diperoleh dari buku-buku ilmiah, laporan penelitian, karangan-karangan ilmiah, tesis dan disertasi, peraturan-peraturan, ketetapan-ketetapan, buku tahunan, ensiklopedia, dan sumber-sumber tertulis baik tercetak maupun elektronik lain.(Purwono, 2009).

2.14.2 Studi Lapangan

2.14.2.1 Pengamatan (*Observasi*)

Observasi adalah pengamatan langsung suatu kegiatan yang sedang dilakukan. Dengan mengamati secara langsung proses atau kegiatan penjualan dari suatu perusahaan.(Jogiyanto, 2005).

2.14.2.2 Wawancara (*Interview*)

Wawancara (*interview*) telah diakui sebagai teknik pengumpulan data atau fakta (*fact finding technique*) yang penting dan banyak dilakukan dalam

pengembangan sistem informasi. Mengadakan tanya jawab sesuai dengan daftar pertanyaan yang telah disusun kepada fungsi yang bersangkutan (Jogiyanto, 2005).

2.14.3 Studi Literatur Sejenis

Mempelajari buku hasil penelitian sejenis sebelumnya yang pernah dilakukan orang lain. Tujuannya ialah untuk mendapatkan landasan teori mengenai masalah yang akan diteliti. Teori ini merupakan pijakan bagi peneliti untuk memahami persoalan yang diteliti dengan benar dan sesuai dengan kerangka berfikir ilmiah (Jomathan, 2006).

2.15 Pengembangan Sistem

Didalam pengembangan sistem ini akan menjelaskan tentang definisi pengembangan sistem, tahapan pengembangan sistem dan konsep siklus pengembangan sistem.

2.15.1 Definisi Pengembangan Sistem

Pengembangan sistem (*system development*) dapat berarti menyusun suatu sistem baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem yang lama perlu diperbaiki atau diganti (Jogiyanto, 2005).

2.15.2 Tahapan Pengembangan Sistem

Proses pengembangan sistem terdiri dari proses standar atau langkah yang dapat digunakan pada semua proyek pengembangan sistem. Meskipun proses

bisnis pada masing-masing organisasi berbeda, mereka memiliki karakteristik umum yang sama, yaitu kebanyakan proses pengembangan sistem pada organisasi mengikuti pendekatan *problem-solving*. Berikut ini adalah langkah *problem-solving* secara umum:

1. Analisis Sistem

Analisis sistem (*system analysis*) dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya (Jogiyanto, 2005).

Tahap analisis sistem dilakukan setelah tahap perencanaan sistem (*system planning*) dan sebelum tahap desain sistem (*system design*). Tahap analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan di dalam tahap ini akan menyebabkan juga kesalahan di tahap selanjutnya (Jogiyanto, 2005).

Didalam tahap analisis terdapat langkah-langkah dasar yang harus dilakukan oleh analis sistem sebagai berikut (Jogiyanto, 2005):

- a) *Identify*, yaitu mengidentifikasi masalah.
- b) *Understand*, yaitu memahami kerja dari sebuah sistem yang ada.
- c) *Analyze*, yaitu menganalisis sistem.
- d) *Report*, yaitu membuat laporan dari hasil analisa.

2. Desain Sistem

Desain sistem dapat diartikan sebagai berikut (Jogiyanto, 2005):

- a) Tahap setelah analisis dari siklus pengembangan sistem.

- b) Pengidentifikasian dari kebutuhan-kebutuhan fungsional.
- c) Persiapan untuk merancang bangunan implementasi.
- d) Menggambarkan bagaimana suatu sistem di bentuk.
- e) Penggambaran berupa, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi.
- f) Konfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem.

Desain sistem (*system design*) dapat dibagi dalam dua bagian, yaitu desain sistem secara umum (*general system design*) dan desain sistem terinci (*detailed system design*). Desain sistem secara umum (*general system design*) disebut juga dengan desain konseptual (*conceptual design*) atau desain logikal (*logical design*) atau desain secara makro (*macro design*). Desain sistem terinci disebut juga dengan desain sistem fisik (*physical system design*) atau desain internal (*internal design*). (Jogiyanto, 2005)

Tujuan dari desain secara umum adalah untuk memberikan gambaran secara umum kepada *user* tentang sistem yang baru. Desain secara umum mengidentifikasikan komponen- komponen sistem informasi yang akan di desain secara rinci. Desain terinci dimaksudkan untuk pemogram komputer dan ahli teknik lainnya yang akan mengimplementasi sistem (Jogiyanto, 2005).

3. Seleksi Sistem

Menyeleksi atau memilih teknologi untuk sistem informasi merupakan tugas yang tidak mudah. Tahap seleksi sistem (*system selection*) merupakan tahap

untuk memilih perangkat keras dan perangkat lunak untuk sistem informasi (Jogiyanto, 2005).

4. Implementasi Sistem

Tahap implementasi sistem (*system implementation*) merupakan tahap meletakkan sistem supaya siap untuk dioperasikan. Tahap ini termasuk juga kegiatan menulis kode program jika tidak digunakan paket perangkat lunak aplikasi dan pengetesan program (Jogiyanto, 2005).

2.16 Literatur Sejenis

1. Judul skripsi “Analisa dan perancangan Basis Data Pada Aplikasi IT Help Desk Berbasis Web di PT. Bank Mandiri Persero. Oleh Mayshe, Anita Mulyana, Oktavia (Universitas Bina Nusantara Jakarta 2007).

Saat ini tela banyak perusahaan-perusahaan besar yang menerapkan aplikasi *report* untuk mengetahui kinerja perusahaan. *Report* tersebut dapat sangat bermanfaat bagi perusahaan dengan cara melaporkan apa saja masalah yang muncul, jumlah masalah serta detailnya, sehingga perusahaan dapat mencari cara mangantisipasinya suatu waktu. *Report* juga dapat membantu pihak manajemen untuk melihat perkembangan perusahaan.

Hal ini pulalah yang akan di implementasikan oleh IT *HelpDesk* PT. Bank Mandiri Persero, jakarta. IT *HelpDesk* berfungsi sebagai solusi permasalahan oleh semua cabang, apabila cabang mengalami masalah yang terhubung dengan *core banking* dan *non care banking*, cabang dapat menghubungi IT *HelpDesk* yang berpusat di jakarta.

2. Judul skripsi “*Anaaisis Dan Perancangan E-Client Relationship Management PT. Rajawali Nusindo*” oleh Farah Nurfuadia (UIN Syarif Hidayatullah 2006).

Menyikapi persaingan yang semakin kompetitif pada setiap bisnis, memulculkan ide untuk memadukan antara teknologi informasi dengan bisnis. Pemesanan secara virtual menggambarkan cakupan yang luas mengenai teknologi informasi, proses dan praktek dalam transaksi bisnis online tanpa menggunakan alat transaksi manual. Dengan menggunakan pemesanan secara online, perusahaan dapat memasarkan suatu produk atau jasa kepada konsumen dengan jangkauan ke seluruh dunia, sehingga dari segi bisnis merupakan peluang yang baik untuk memperluas pangsa pasar dari produk atau jasa yang ditawarkan.

Penelitian skripsi ini bertujuan untuk menganalisa dan membuat aplikasi sistem pemesanan produk berbasis web pada PT. Rajawali Nusindo yang dapat menyajikan informasi yang dibutuhkan konsumen. Dengan sistem ini perusahaan dapat membangun sebuah sistem pemesanan produknya berbasis internet yang bisa menghilangkan keterbatasan jarak dan waktu.

Alat pengembangan yang digunakan peneliti yaitu PHP sebagai aplikasi interface dan Mysql sebagai basis datanya. Ini semua dipilih karena kehandalan dalam membuat aplikasi berbasis *web*. Metode pengembangan yang digunakan adalah *System Development Life Cycle (SDLC) Waterfall*.

3. Judul skripsi “*Pengembangan Sistem Monitoring Help desk pada Pustekkom Depdiknas*” oleh Lia Via Pratiwi (UIN. Syarif Hidayatullah Jakarta 2006).

PUSTEKKOM (Pusat Teknologi Informasi dan Komunikasi Pendidikan) sebagai sebuah lembaga yang langsung berada di bawah Menteri Pendidikan Nasional dan bertugas melaksanakan, mengkoordinasikan dan membina kegiatan di bidang teknologi informasi dan komunikasi pendidikan. Unit Tata Kelola TIK (Teknologi Informasi dan Komunikasi) DEPDIKNAS tingkat pusat di pimpin oleh seorang ketua yang bertanggung jawab langsung kepada Direktur DEPDIKNAS. Terdapat 3 (tiga) bidang di struktur unit tersebut yaitu: Bidang Perencanaan dan Regulasi, Bidang Operasional Teknis dan Bidang Evaluasi dan *Monitoring*. Ketiga bidang tersebut bersama-sama mengelola layanan utama TIK DEPDIKNAS yaitu: Layanan Infrastruktur Jaringan dan *Internet*, Layanan Aplikasi dan Konten serta Layanan Dukungan Pengguna. Pengelolaan dukungan pengguna TIK DEPDIKNAS menjadi sangat penting karena merupakan proses terdepan dari *Client satisfaction* pada seluruh layanan TIK yang disediakan oleh DEPDIKNAS. Layanan dukungan pengguna tersebut meliputi: *Call Center / Helpdesk*, Pusat Otentifikasi, Otorisasi dan Identifikasi Pengguna, Pelatihan Operasional Pengguna. Dimana *Help desk* akan berfungsi sebagai solusi untuk menangani permasalahan yang ada, khususnya LAN (*Local Area Network*) yang ada pada PUSTEKKOM DEPDIKNAS. LAN PUSTEKKOM bertanggung jawab terhadap segala konektivitas jaringan

internet yang ada di lingkungan kerja PUSTEKKOM DEPDIKNAS, sehingga aktivitas jaringan tersebut dapat berjalan dengan baik, lancar dan tak ada gangguan. Adapun masalah yang berkaitan dengan kerusakan *hardware*, tim *support help desk* akan menyampaikan kebutuhan ke Sub.Bidang Rumah Tangga, apabila memungkinkan untuk langsung menyelesaikan masalah dan hasilnya akan di terima oleh *font desk*. Namun penanganan masalah yang saat ini terjadi pada LAN PUSTEKKOM masih kurang baik, karena tidak terkontrolnya proses dalam memperbaiki koneksi jaringan LAN sehingga membuat pegawai menunggu dan tidak terselesaikannya kerusakan yang ada. Belum adanya sistem pada PUSTEKKOM DEPDIKNAS yang memberikan laporan secara sistematis. Semua pencatatan laporan, laporan bulanan serta perhitungan masalah masih dilakukan manual. Hal ini dapat memberi dampak negatif, yakni: terdapat masalah yang beresiko di catat secara berulang, adanya masalah yang tidak tercatat karena sibuknya operator dalam menerima telepon, perhitungan masalah tidak akurat karena dilakukan manual dan perlunya Kepala Bidang IT menunggu untuk mendapatkan laporan yang diinginkan. Dan juga belum adanya sistem yang mampu mengukur tingkat masalah pada *help desk* dalam menangani permasalahan yang terjadi pada LAN PUSTEKKOM.

BAB III

METODOLOGI PENELITIAN

3.1 Metode Pengumpulan Data

Dalam pelaksanaan penelitian ini diperlukan data dan informasi terkait yang nantinya digunakan sebagai bahan rujukan untuk pengembangan sistem yang ada serta mendukung keabsahan pembahasan pada laporan penelitian. Adapun metode pengumpulan data yang digunakan dalam penelitian ini adalah:

3.1.1 Studi Pustaka

Pada metode ini peneliti mempelajari buku-buku yang berhubungan dengan masalah yang dibahas, serta mencari informasi-informasi juga di internet yaitu mencari teori-teori yang berhubungan dengan permasalahan yang akan peneliti kembangkan. diantaranya membaca buku dari Pengenalan Sistem Informasi oleh Abdul Kadir, Analisis dan Design Sistem Informasi oleh Jogiyanto, Analisis dan Desain Sistem Informasi oleh Al-Bahra Bin Ladjamudin, Pengenalan Komputer oleh Yogiyanto, beberapa situs internet seperti www.ilmukomputer.com, www.indoskripsi.com. Selebihnya ada pada daftar pustaka.

3.1.2 Studi Lapangan

a. Observasi

Pengamatan dilakukan dengan melakukan peninjauan langsung PT. Primus Indojoya terutama pada bidang IT *Helpdesk*, hal-hal yang diamati adalah proses *Helpdesk Ticketing System* yang sedang

berjalan dan juga kegiatan-kegiatan yang berhubungan dengan *Helpdesk*. Tujuan teknik ini adalah untuk mengumpulkan data dan informasi. Observasi dilaksanakan PT. Primus Indojoya, beralamat di Jl. Gading Bukit Indah Blok A/2, kelapa gading. dan membutuhkan waktu ± 3 bulan, dari bulan April sampai dengan Juni 2011. Dari hasil observasi didapat data-data dan informasi yang digunakan dalam membangun *helpdesk ticketing system*. Selengkapnya ada pada *lampiran 1*.

b. Wawancara

Wawancara dilakukan dengan kepala bagian IT *Helpdesk* yaitu bapak Alex Kurniawan untuk memperoleh data-data dan informasi yang dibutuhkan dalam perancangan *Helpdesk Ticketing System*. Tujuan wawancara ini adalah untuk mendapatkan informasi yang dibutuhkan dalam pengembangan *Helpdesk Ticketing System*. Wawancara dilakukan pada tanggal 04 April 2011. Pertanyaan yang diajukan sebanyak 5 pertanyaan dan setiap pertanyaan dijawab langsung oleh bapak Alex Kurniawan. Berdasarkan hasil wawancara yang peneliti lakukan, didapat profile perusahaan PT. Primus Indojoya dan juga informasi yang berguna dalam perancangan *helpdesk ticketing system*. Daftar pertanyaan wawancara dapat dilihat pada *lampiran 2*.

3.1.3 Studi Literatur Sejenis

Metode studi literatur yang peneliti lakukan adalah dengan membandingkan *research* sejenis. Perbandingan dilakukan pada sistem *helpdesk*,

diantaranya itu adalah Analisa dan perancangan basis data pada aplikasi IT *Help Desk* berbasis web pada PT. Bank Mandiri Persero yang dibandingkan dengan Rancang bangun *helpdesk ticketing system* dengan pendekatan Client relationship management (CRM) yang dibuat oleh peneliti, sehingga peneliti bisa mengetahui hal-hal apa saja yang belum dilakukan dalam penelitian sebelumnya dan terhindar dari kesalahan-kesalahan dalam penelitian sebelumnya. Sebagai referensi ada di akhir bab 2.

3.2 Metode Pengembangan Sistem

Penelitian ini menggunakan metode *iteration waterfall* dengan langkah-langkah penyelesaian masalah yang dikemukakan dalam buku *Metode Desain dan Analisis Sistem* yang ditulis oleh Jeffery L. Whitten, Lonnie D. Bentley, dan Kevin C. Dittman (2004).

Terdapat empat langkah pengembangan sistem yang di dalamnya terdapat langkah-langkah penyelesaian masalah dalam penelitian *helpdesk ticketing system*.

3.2.1 Permulaan Sistem (*System Initiation*)

Pada tahap ini ada beberapa poin penting yang perlu dilakukan dalam pembuatan Rancang Bangun *HelpDesk Ticketing System*, yaitu identifikasi masalah, penetapan lingkup sistem, penetapan tujuan pengembangan sistem, dan pembuatan jadwal pengembangan sistem.

1. Identifikasi Masalah

Melakukan identifikasi terhadap masalah yang terjadi pada *helpdesk* yang sedang berjalan dalam hal ini sistem *helpdesk* yang digunakan masih manual dalam proses penginputan data dan informasi.

Permasalahan yang ditemukan dalam identifikasi masalah kemudian digunakan untuk merencanakan solusi yang tepat untuk pemecahan masalah tersebut.

2. Lingkup Sistem

Menentukan batasan ruang lingkup sistem yang akan dikembangkan dalam *helpdesk ticketing system*, di antaranya lingkup pembuatan aplikasi pada IT *helpdesk*, pembuatan laporan bulanan.

Pada *helpdesk ticketing system* ini, hanya membahas sebatas proses IT *helpdesk ticketing system* saja, kemudian dari data dan informasi yang diperoleh dikembangkan agar menghasilkan laporan yang akurat.

3. Tujuan Pengembangan Sistem

Menentukan tujuan pengembangan sistem, dengan menentukan untuk apa *helpdesk ticketng system* ini dikembangkan dan siapa saja pengguna dari sistem ini.

4. Jadwal Pengembangan Sistem

Pembuatan jadwal pengembangan sistem, yaitu membuat alokasi waktu untuk keseluruhan pembuatan sistem, langkah demi langkah sesuai metode pengembangan sistem *iteration waterfall*, mulai dari permulaan sistem (*system initiation*) sampai implementasi sistem (*system implementation*).

3.2.2 Analisis Sistem (*System Analysis*)

Pada tahap ini, akan dilakukan beberapa tahapan, yaitu mengenal instansi terkait dalam hal ini PT. Primus Indojoya, melakukan analisis dan pemahaman terhadap masalah, melakukan analisis persyaratan sistem yang akan dikembangkan, dan membuat analisis pemodelan berorientasi obyek terhadap sistem yang sedang berjalan.

1. Gambaran Umum PT. Primus Indojoya

Menguraikan tentang profil PT. Primus Indojoya dengan singkat.

2. Sistem yang Berjalan

Helpdesk PT. Primus Indojoya berfungsi untuk memberikan informasi dan penanganan gangguan apabila terjadi kendala dalam penggunaan Internet.

- a. Setiap informasi atau masalah dapat disampaikan kepada *helpdesk* melalui empat cara, yaitu: *Via telephone*, *Via SMS*, *Via Email*, *fax*.
- b. Kekurangan dari sistem yang berjalan saat ini yakni:
 1. Proses penanganan masalah koneksi jaringan internet dan kerusakan *hardware* tidak terkontrol menjadikan permasalahan yang tidak tertangani.
 2. Belum adanya laporan secara sistematis merupakan salah satu faktor bagi Kepala Bidang IT untuk evaluasi hasil laporan dari permasalahan yang ada.

3. Analisis Sistem

Dari analisis masalah yang ada, maka dibutuhkan suatu sistem yang mampu memonitor permasalahan yang terjadi sehingga secara *real time* masalah dapat tertangani dengan baik.

Database yang terintegrasi dengan baik membuat suatu perhitungan masalah untuk memberikan laporan yang akurat.

Kebutuhan pengguna diidentifikasi, yakni: *database* yang terintegrasi dengan baik, pencatatan masalah yang tersistem, laporan bulanan yang akurat, informasi permasalahan yang terjadi, penanganan masalah yang jelas oleh siapa yang menanganinya.

Oleh karena itu, penulis memberikan alternatif solusi sebagai pemecahan masalah akan kebutuhan pengguna, yakni:

- a. Sistem yang memiliki hak akses kepada pihak tertentu untuk dapat memperoleh informasi tentang permasalahan yang ditangani oleh *helpdesk*, merancang sistem secara *real time* pada pencatatan masalah.
- b. Sistem dengan hasil laporan yang akurat untuk membantu Kepala Bidang IT dalam evaluasi hasil laporan serta mengukur tingkat masalah pada *helpdesk* dalam penanganan dari permasalahan yang ada baik terkait dengan koneksi jaringan internet maupun kerusakan *hardware*.

Kelebihan dari sistem yang akan dikembangkan yakni dapat memberikan informasi sesuai dengan kebutuhan pengguna, pemecahan masalah yang *real time*, dan keakuratan laporan bulanan. Kekurangan dari sistem yang akan dikembangkan yakni keamanan data yang belum terkondisikan.

4. Usulan Sistem Baru

Sistem yang akan dikembangkan dapat membantu pihak *helpdesk* untuk memantau adanya permasalahan yang timbul dan langsung melakukan *input*

pencatatan data masalah atau mendatangi pelapor untuk menangani masalah tersebut.

Masalah yang ditangani oleh bagian *helpdesk* dilakukan secara *online* dengan menggunakan internet PT. Primus Indojaya apabila ada kerusakan *hardware*, *software* pada client kemudian *helpdesk* mengirim pesan kepada Sub.Bidang IT untuk menyelesaikan masalah kemudian data masuk dalam *database*.

Client dapat langsung melaporkan masalah melalui *inteface* sistem yang disediakan tanpa harus menemui langsung, hasil pencatatan dan perhitungan masalah akan digunakan sebagai laporan kepada Kepala Bidang IT untuk menentukan kebijakan selanjutnya. Alur usulan sistem baru akan dijelaskan selengkapnya pada BAB IV.

3.2.3 Desain Sistem (*System Design*)

Pada tahap ini akan dilakukan perancangan dari perangkat lunak *helpdesk ticketing system* berdasarkan hasil analisa yang telah dilakukan sebelumnya.

1. Perancangan Proses

Perancangan proses di sini dimaksudkan untuk membuat pemodelan terhadap sistem baru yang dapat mewakili sistem yang berjalan saat ini pada *helpdesk*. Dengan 4 entitas yang terkait yakni Admin (*frontdesk*), Tim *Support*, kerusakan *hardware*, dan Kepala Bidang IT. Perancangan proses akan dijelaskan dengan pemodelan UML pada BAB IV.

2. Perancangan Database

Penulis menyimpulkan bahwa diperlukannya identifikasi tipe entitas yakni terkait pegawai, subbidrt, admin, timsupport, kepbidit, mslh, jenis_mslh, status, mslh_hd, mslh_lan. Database yang akan di bangun menggunakan tools: ERD, normalisasi, kamus data, struktur data, dan STD (*State Transition Diagram*). Keterangan simbol-simbol yang digunakan penulis terlampir dalam daftar simbol.

3. Perancangan Tampilan

Untuk tampilan antarmuka (*interface*) pada sistem ini, penulis melakukan perancangan tampilan yang terdiri dari rancangan masukan dan rancangan keluaran.

4. Perancangan Jaringan

Untuk perancangan arsitektur jaringan pada pengembangan *Helpdesk ticketing system* ini menggunakan jaringan Internet, informasi dapat di peroleh secara menyeluruh.

3.2.4 Implementasi Sistem (*System Implementation*)

Pada tahap ini akan dilakukan konstruksi perangkat lunak, pengevaluasian terhadap hasil perancangan *helpdesk ticketing system* dan perangkat lunak yang dikembangkan dan pengimplementasian terhadap sistem tersebut.

1. Coding

Pada pengembangan sistem ini, penulis menggunakan bahasa pemrograman *Hypertext Processor* (PHP) dengan menggunakan database mysql.

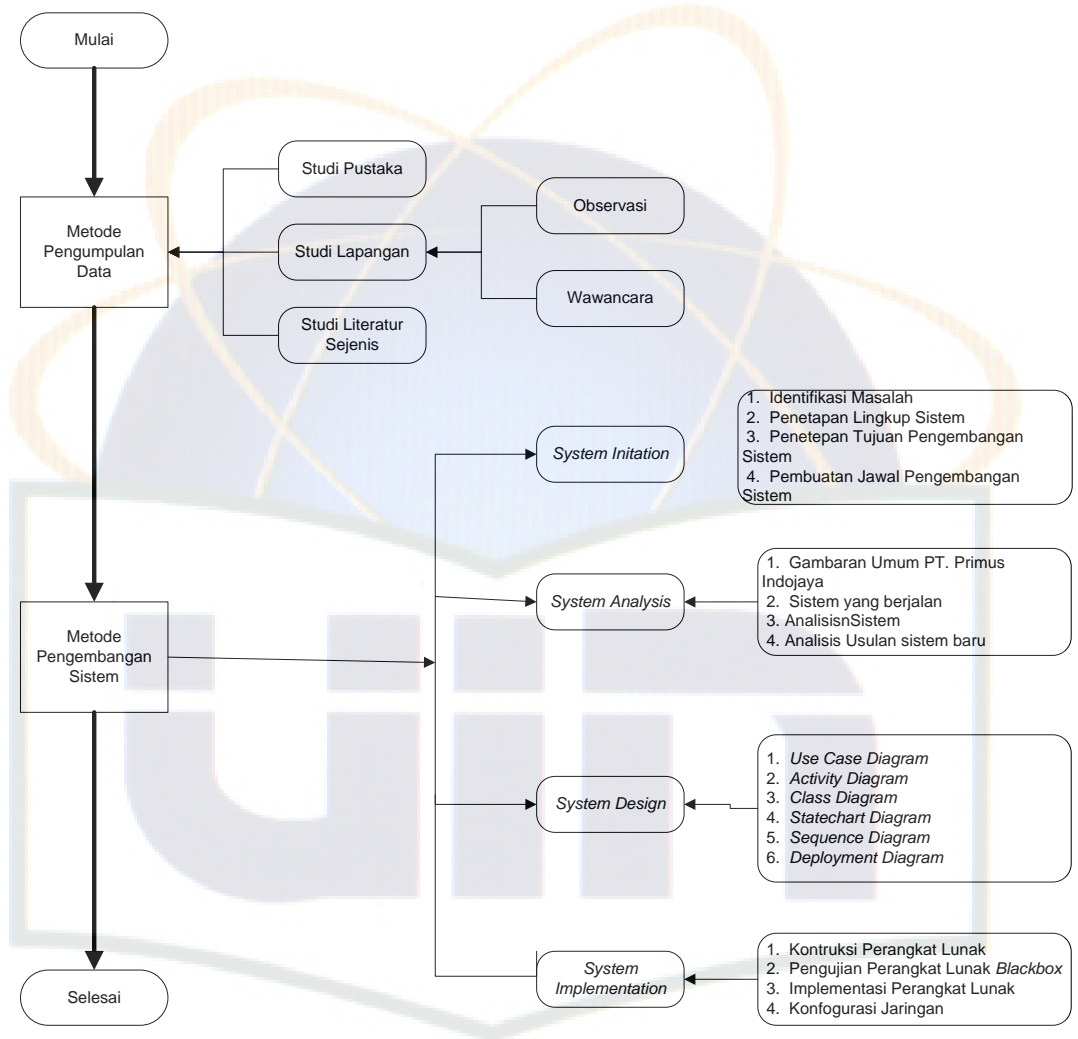
2. Pengujian Sistem

Pengujian *helpdesk ticketing system* yang telah dikonstruksikan, kemudian diuji agar dapat mengetahui kesalahan-kesalahan yang terdapat dalam perangkat lunak sistem dan kemudian dapat diperbaiki agar sesuai dengan yang diharapkan.

Pengujian tersebut dilakukan dengan dua tahap yaitu tahap *alpha testing* dengan menggunakan data *dummy* (tidak sebenarnya) dan tahap *beta testing* menggunakan data sebenarnya yang terdapat pada PT. Primus Indojoya. Pengujian dilakukan menggunakan metode *black box testing* yaitu dengan cara meng-inputkan data ke dalam sistem dan melihat hasil *outputnya* apakah sesuai dengan yang diharapkan.

3.3 Kerangka Berfikir Penelitian

Dalam melakukan penelitian ini, peneliti melakukan tahapan-tahapan kegiatan dengan mengikuti rencana kegiatan yang tertuang dalam kerangka berpikir meliputi metode pengumpulan data dan metode pengembangan sistem. Gambar 3.1 merupakan kerangka berpikir penelitian sistem ini.



Gambar 3.1 Kerangka berpikir Penelitian

BAB IV

HASIL DAN PEMBAHASAN

4.1 *System Initiation* (Permulaan Sistem)

4.1.1 Identifikasi Masalah

Dari hasil wawancara dengan staf *helpdesk ticketing system*, penulis mendapatkan beberapa permasalahan terhadap prosedur penanganan dan data yang ada sehingga menghambat kinerja staf *helpdesk ticketing system*. Masalah yang muncul antara lain:

- a. Belum adanya *database* yang terintegrasi dengan sistem.
- b. Pencatatan masalah masih manual (dalam *Microsoft Excel*) dan tidak melalui sistem tertentu.
- c. Laporan bulanan dihitung secara manual (kemungkinan data yang tidak akurat).

Kebutuhan pengguna diidentifikasi, yakni: *database* yang terintegrasi dengan baik, pencatatan masalah yang tersistem, laporan bulanan yang akurat, informasi permasalahan yang terjadi, penanganan masalah yang jelas oleh siapa yang menangani.

Untuk memfokuskan penelitian yang dilakukan, maka penulis akan membatasi lingkungan penelitian pada *helpdesk ticketing system* yang berada pada PT.PRIMUS INDOJAYA. *Helpdesk ticketing system* memiliki tugas untuk membantu pegawai atas masalah mereka terutama keluhan yang berhubungan dengan jaringan komputer maupun sistem.

Tujuan *Helpdesk ticketing system ini*, yakni mampu memberikan informasi atas permasalahan yang terjadi baik koneksi jaringan internet maupun kerusakan *hardware, software* serta memberikan analisis yang tepat yang akan digunakan sebagai indikator dalam pengambilan keputusan untuk menindaklanjuti penanganan masalah.

4.1.2 Penetapan Lingkup Sistem

Berdasarkan identifikasi masalah yang telah dilakukan, maka akan dikembangkan suatu *Helpdesk ticketing system* pada PT. Primus Indojoya dengan batasan-batasan pengembangan seperti: Sistem ini dikembangkan menggunakan *server Apache*, bahasa pemrograman PHP, *database MySQL*, dijalankan melalui *browser* dan memerlukan sambungan internet. Selain itu, pada penelitian ini tidak membahas jaringan komunikasi serta keamanan data yang terkait dengan sistem ini secara terinci.

4.1.3 Penetapan Tujuan Pengembangan Sistem

Tujuan *helpdesk ticketing system ini*, yakni mampu memberikan informasi atas permasalahan yang terjadi baik koneksi jaringan internet maupun kerusakan *hardware, software* serta memberikan analisis yang tepat yang akan digunakan sebagai indikator dalam pengambilan keputusan untuk menindaklanjuti penanganan masalah.

Selain itu, pengembangan sistem ini dimaksudkan guna membantu operator dalam penginputan data yang selama ini masih menggunakan sistem

manual dengan bantuan aplikasi *Microsoft Excel* sebagai perangkat penyimpanan data.

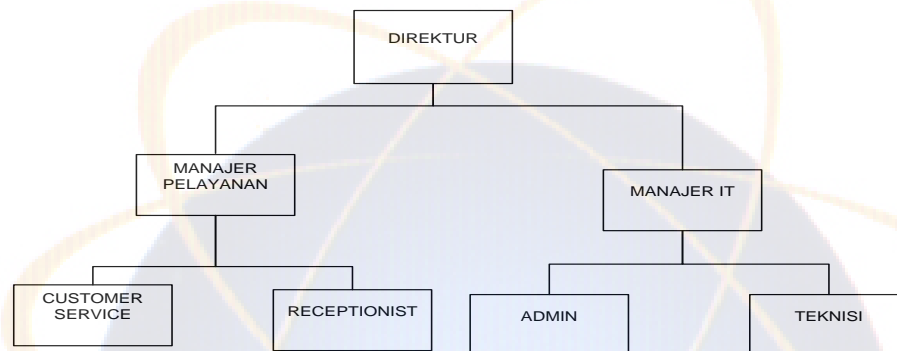
4.2 System Analysis (Analisis Sistem)

4.2.1 Gambaran Umum PT. Primus Indojoya

PT. Primus Indojoya adalah perusahaan yang terintegrasi, terdistribusi dan terlengkap dalam memberikan informasi teknologi bisnis, yang berkaitan dengan masalah komputer bisnis, jaringan, data komunikasi, *software development* dan *supplies peripherals*. Perusahaan memiliki banyak konsultan, *outsourcing*, *intranet* dan solusi *internet*, sistem *design* dan *engineering*, komputer aksesoris *supplies*, penginstalan, *training* dan servis *maintenance*. Perusahaan ini juga mempunyai *client* atau *partner* kerja yang berkompeten, yang saling memberikan keuntungan diantara keduanya.

PT. Primus Indojoya beralamat di Jl. Kelapa gading Boulevard 2 Kav. 3 Jakarta Utara. PT. Primus Indojoya dapat menangani masalah yang berkaitan dengan *helpdesk ticketing system* pada PT. Primus Indojoya dapat berfungsi sebagai solusi permasalahan dari *client* yang mengalami masalah. Masalah tersebut dicatat oleh operator dan dikumpulkan dalam suatu basis data untuk di *input* ke dalam sistem. Perkembangan PT. Primus Indojoya, sampai saat ini semakin membaik, banyak perusahaan yang masih berlangganan dan banyak juga perusahaan yang baru bergabung.

4.2.2 Struktur Organisasi



Gambar 4.1 Struktur Organisasi PT. Primus Indojoya

4.2.3 Analisis Sistem yang Berjalan

Pada dasarnya, *Helpdesk ticketing system* PT. Primus Indojoya berfungsi sebagai *front end* informasi kepada pengguna dan solusi untuk mengatasi permasalahan yang dikeluhkan oleh *Client*. Masalah yang dilaporkan dapat berupa koneksi jaringan internet dan kerusakan *hardware, software* pada *client* terkait pelaporan masalah tersebut. Unit kerja *Helpdesk* terdiri dari 2 tim (*frontdesk* dan tim *support*) dengan jumlah pegawai 20 orang, dimana 15 pegawai tetap dan 5 pegawai honorer. *Helpdesk ticketing system* mengelola 2 tugas: pencatatan dan penyelesaian masalah pada unit yang terkait dan penyedia laporan.

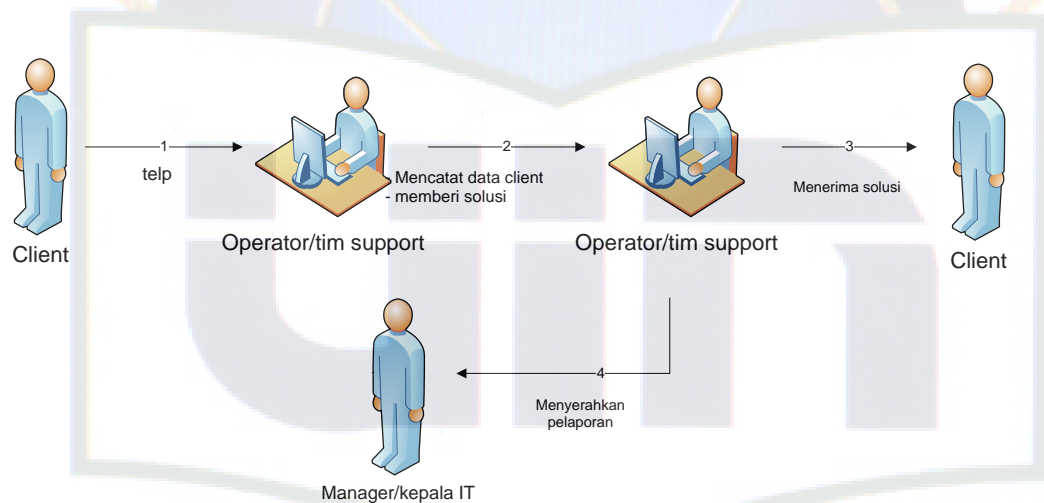
Ada beberapa prosedur yang harus dilakukan oleh pegawai pada saat melaporkan masalah. Prosedur yang harus dilakukan terkait dengan operator (*frontdesk* dan tim *support*). Pihak terakhir yang terkait dengan masalah ini adalah Kepala Bidang IT. Tugas dan tanggung jawab masing-masing pihak adalah sebagai berikut:

- a. Operator (*Frontdesk* dan Tim *Support*)

Berfungsi sebagai penerima dan pencatat dari permasalahan pada unit yang terkait. Pada saat yang sama, operator akan memberikan solusi atas masalah tersebut *Via telephone*.

b. Kepala Bidang IT

Kepala Bidang IT sebagai pihak yang menerima laporan tiap bulan, dimana hasil dari laporan tersebut dapat digunakan untuk mengambil keputusan.



Gambar 4.2 *Standart Operation Prcsedure (SOP) Sistem yang berjalan*

Berikut ini adalah penjelasan *helpdesk ticketing system* yang sedang berjalan seperti yang digambarkan sebelumnya:

1. *Client* menelpon operator untuk memberitahukan keluhan dan masalah yang dialami.
2. Operator mencatat masalah *Client* untuk diinput ke dalam sistem.
3. Kemudian *Client* menerima solusi dari operator.
4. Kemudiann operator menyerahkan laporan pencatatan masalah dan di simpan oleh manager IT.

4.2.4 Analisis Kelebihan dan Kekurangan Sistem yang Berjalan

Berdasarkan analisis sistem yang berjalan pada *Helpdesk Ticketing System* pada PT. Primus Indojoya, dapat diuraikan beberapa kelebihan dan kekurangan pada proses sistem yang berjalan secara umum.

1. Kelebihan Sistem

- a. Penyelesaian masalah *Client* cepat, karena langsung dijawab melalui telpon oleh tim support.
- b. Pencatatan keluhan dan masalah sudah menggunakan komputer, hanya saja meski terbatas pada penyimpanan data dan pencetakan laporan.

2. Kekurangan Sistem

- a. Pencatatan masalah yang menggunakan *software microsoft excel* masih belum mempunyai aturan yang baku.
- b. Penyimpanan data masih acak, sehingga susah dalam pencarian *record* dan pelaporan.
- c. Hasil laporan ke manajer atau kepala bagian IT sering terlambat, karena masih di buat secara manual.

4.2.4.1 Analisa Perbandingan Sistem

Perbandingan sistem berjalan dengan sistem usulan diklasifikasikan berdasarkan kekurangan dan kelebihan dapat dilihat pada tabel 4.2.

Tabel 4.1 Analisa Perbandingan Sistem

Keterangan	Kekurangan	Kelebihan
Sistem Berjalan	- Pencatatan masalah masih menggunakan Microsoft excel masih belum mempunyai aturan	

	<p>yang baik.</p> <ul style="list-style-type: none"> - Proses laporan ke manager sering terlambat karena sistem masih manual. - Proses dokumentasi pencacatan dan penyimpanan belum baik. - Keamanan data yang belum terkondisikan. 	
Sistem Usulan		<ul style="list-style-type: none"> - Proses pencatatan sudah terkomputerisasi dan langsung masuk ke sistem database. - Penyelesaian masalah Client cepat, karena langsung di jawab melalui telpon. - Proses dokumentasi menjadi mudah dan aman. - Dapat memberikan informasi sesuai kebutuhan pengguna. - Pemecahan masalah yang realtime, dan keakuratan laporan bulanan.

4.2.4.2 Sistem Literatur Sejenis

Dari analisa literatur sejenis maka penulis mencoba membandingkan kelebihan sistem yang dibuat penulis dengan kekurangan sistem yang lama dari literatur sejenis. Berikut tabel perbandingannya:

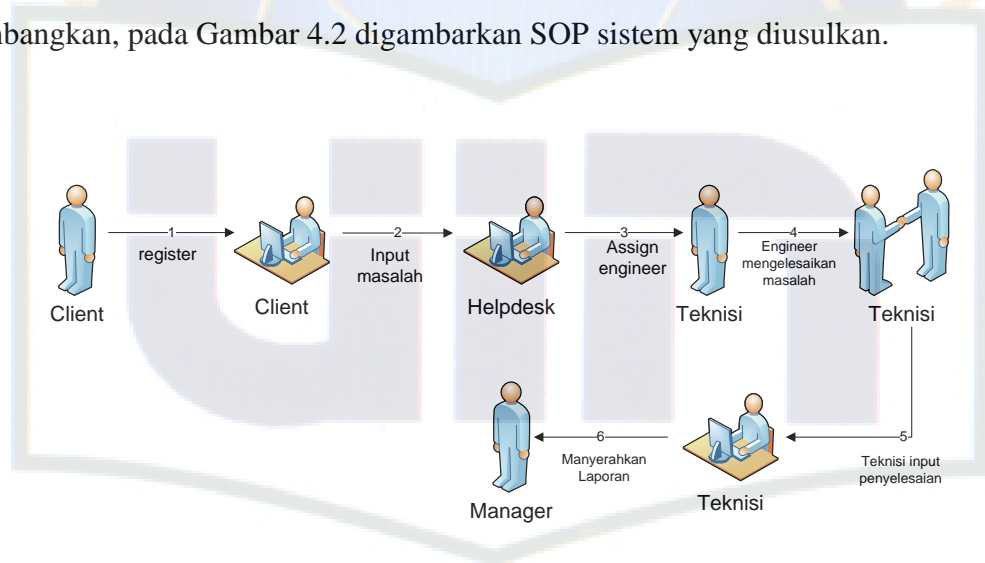
Tabel 4.2 Perbandingan Literatur Sejenis

No	Literatur Sejenis	Sistem Usulan	Sejenis/kesamaan
1	Analisa sistem belum diimplementasikan ke dalam bentuk aplikasi	Analisa sistem sudah diimplementasikan kedalam bentuk sistem aplikasi	Bagian yang sejenis dalam kedua sistem ini yaitu dari permasalahan jarak antara gudang dengan management Sistem yang sama mengenai pengolahan

			data barang masuk dan masuk (bener data masuk dan masuk apa data masuk dan keluar?)
--	--	--	---

4.2.5 Analisis Usulan Sistem Baru

Mereferensi dari kelebihan dan kekurangan sistem yang saat ini sedang berjalan pada PT. Primus Indojoya serta kebutuhan sistem yang akan dikembangkan, pada Gambar 4.2 digambarkan SOP sistem yang diusulkan.



Gambar 4.3 *Standart Operation Procedure (SOP) Sistem Usulan*

Berikut ini akan adalah penjelasan tentang SOP *helpdesk Ticketing System* yang diusulkan.

1. Client datang atau menelpon untuk mendaftar atau melakukan register.
2. Kemudian operator mencatat dan menginput masalah kedalam sistem.
3. Kemudian helpdesk memberika tugas kepada engineer.
4. Teknisi datang ke perusahaan untuk menyelesaikan masalah dan memberikan solusi.

5. Kemudian Teknisi memberitahukan client bahwa permasalahannya telah diselesaikan melalui email atau bertemu langsung dengan client.
6. Kemudian Teknisi memberikan laporan kepada manager bagian IT.

4.2.6 Analisis Kebaikan Sistem yang Diusulkan

Berdasarkan SOP yang diusulkan sebelumnya, berikut ini dilakukan analisis terhadap kebaikan usulan sistem tersebut secara lebih detil, yaitu:

1. Sudut Pandang Bisnis

Ruang lingkup bisnis *HelpDesk Ticketing System* adalah mendukung seluruh kegiatan dalam Penanganan Sistem di perusahaan tersebut.

2. Sudut Pandang Operasional

a. Ketersediaan dan Keandalan.

Helpdesk Ticketing System tersedia bagi pengguna selama 24 jam 7 hari kerja untuk mendukung fungsi-fungsi di dalamnya, khusus untuk menerima dan menjawab keluhan dan masalah dari perusahaan yang terkait.

b. Performansi

Helpdesk Ticketing System menyediakan pemakaian *multi-user* dan dapat tetap digunakan dalam kondisi yang melibatkan jumlah kegiatan dalam sistem dengan jumlah yang banyak tanpa mengganggu stabilitas sistem.

c. Pelatihan Pengguna

Helpdesk Ticketing System dapat digunakan oleh seluruh pengguna dengan melalui pelatihan menggunakan *user manual*.

3. Sudut Pandang Pengguna

a. Akses Pengguna

Helpdesk Ticketing System merupakan sebuah sistem yang berjalan pada jaringan komputer dengan memanfaatkan jaringan internet.

- b. Mudah untuk di gunakan oleh pengguna

Helpdesk Ticketing System mudah digunakan dan *user friendly*.

4. Sudut Pandang Keamanan

Kebutuhan akses kontrol dari *Helpdesk Ticketing System* meliputi:

- a. Seluruh pengguna yang dapat *login* ke dalam *Helpdesk Ticketing System* melalui website yang sudah ada.
- b. Jika pengguna gagal untuk memasukkan *password* yang benar, maka pengguna tidak dapat mengakses sistem.

4.2.7 Analisis dan Pemodelan Sistem Berorientasi Obyek

Berikut ini akan dijabarkan analisis dan pemodelan berorientasi obyek sebagai penggambaran *blue print* (cetak biru) sistem dari seluruh hasil analisis yang telah dilakukan sebelumnya.

4.2.7.1 Pemodelan *Usecase Diagram*

1. Identifikasi Pelaku (*Actor*) Bisnis

Pada Tabel 4.6 dijabarkan daftar pelaku yang terlibat dalam *helpdesk ticketing system* beserta deskripsi tentang masing-masing pelaku yang ada.

Tabel 4.6 Daftar Pelaku (*Actor*) pada *HelpDesk Ticketing System*.

No	Pelaku (<i>Actor</i>)	Deskripsi
1	<i>Client</i>	Orang yang mempunyai keluhan dan masalah.
2	<i>Frondesk / Helpdesk</i>	Orang yang mengelola data sistem, assign

		masalah ke teknisi.
3	<i>Teknisi</i>	Orang yang menyelesaikan masalah.
4	<i>Manajer</i>	Orang yang menerima laporan.

2. Identifikasi *Usecase*

Pada Tabel 4.7 dijabarkan daftar *usecase* yang terlibat dalam *helpdesk ticketing system* beserta deskripsi tentang masing-masing *usecase* dan *actor* yang terlibat dalam *usecase* tersebut.

Tabel 4.7 Daftar *Usecase HelpDesk Ticketing System*

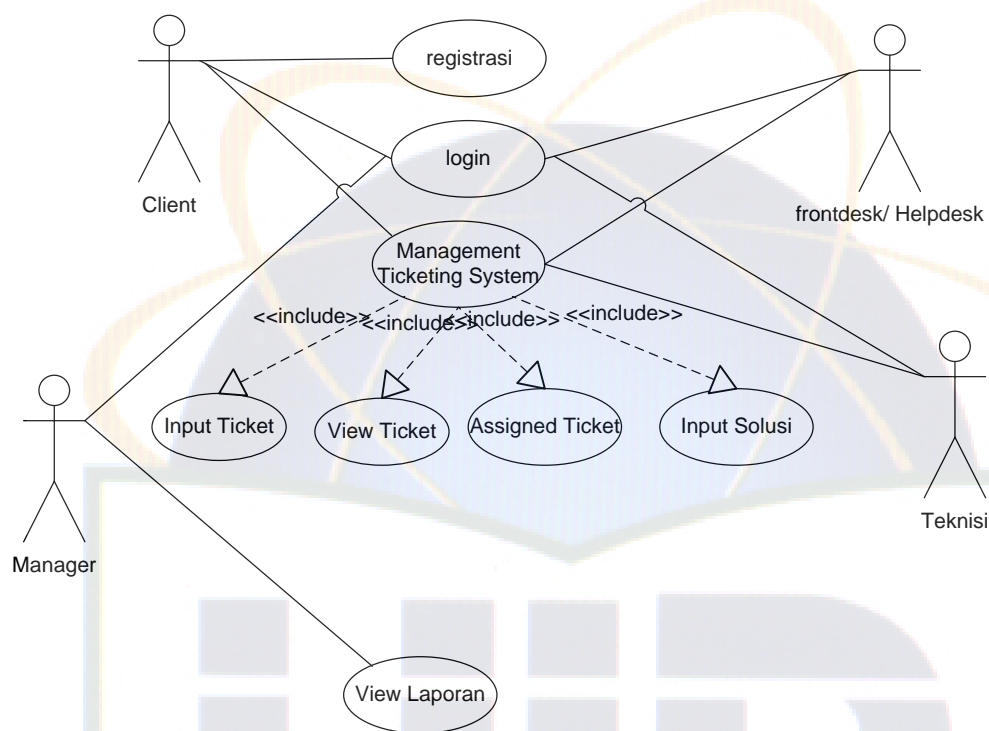
No	Nama <i>Usecase</i>	Deskripsi
1	Register	<i>Usecase</i> menggambarkan kegiatan registrasi Client ke dalam sistem untuk memperoleh account.
2	Login	<i>Usecase</i> menggambarkan kegiatan <i>login</i> ke dalam sistem dengan memasukkan data pengguna untuk diverifikasi oleh sistem.
3	<i>Input Ticket</i>	<i>Usecase</i> menggambarkan kegiatan memasukkan data <i>Ticket</i> .
4	<i>View Ticket</i>	<i>Usecase</i> menggambarkan kegiatan melihat data <i>Ticket</i> .
5	<i>Assiggned Ticket</i>	<i>Usecase</i> menggambarkan kegiatan menugaskan teknisi untuk menyelesaikan masalah atau <i>ticket</i> .
6	Input solusi	<i>Usecase</i> menggambarkan kegiatan memasukkan data solusi atau penyelesaian terhadap <i>ticket</i> kedalam sistem.
7	Lihat Laporan	<i>Usecase</i> menggambarkan kegiatan melihat laporan <i>ticket</i> .

3. Pembuatan *Usecase Model Diagram*

Setelah *usecase* dan *actor* teridentifikasi, *usecase model diagram* digunakan untuk menggambarkan secara grafis lingkup dan batasan sistem.

Tabel 4.8 Daftar Istilah *Usecase Help Desk Ticketing System*

No	Nama <i>Usecase</i>	Deskripsi	Pelaku yang Berpartisipasi
1	Register	<i>Usecase</i> menggambarkan kegiatan registrasi <i>Client</i> ke dalam sistem untuk memperoleh account.	<i>Client</i>
2	Login	<i>Usecase</i> menggambarkan kegiatan <i>login</i> ke dalam sistem dengan memasukkan data pengguna untuk diverifikasi oleh sistem.	<i>Client</i> , <i>frondesk</i> , <i>teknisi</i> , <i>manager</i> .
3	Input Ticket	<i>Usecase</i> menggambarkan kegiatan memasukkan data <i>Ticket</i> .	<i>Client</i>
4	View Ticket	<i>Usecase</i> menggambarkan kegiatan melihat data <i>Ticket</i> .	<i>Client</i> , <i>frondesk</i> , <i>teknisi</i> , <i>manager</i> .
5	Assiggned Ticket	<i>Usecase</i> menggambarkan kegiatan menugaskan teknisi untuk menyelesaikan masalah atau <i>ticket</i> .	<i>Frondesk</i> .
6	Input solusi	<i>Usecase</i> menggambarkan kegiatan memasukkan data solusi atau penyelesaian terhadap <i>ticket</i> kedalam sistem.	<i>Teknisi</i>
7	Lihat Laporan	<i>Usecase</i> menggambarkan kegiatan melihat laporan <i>ticket</i> .	<i>Manager</i> .



Gambar 4.4 Usecase model diagram Help Desk Ticketing System.

4.2.7.2 Narasi Usecase Analysis

Berdasarkan penggambaran *usecase*, Tabel 4.9 – Tabel 4.15 menjelaskan narasi dari *usecase* tersebut.

Tabel 4.9 Narasi Usecase Analysis Register

<i>Use Case Name</i>	<i>Register</i>	
<i>Use Case ID</i>	1	
<i>Actor</i>	Client	
<i>Description</i>	<i>Usecase</i> menggambarkan kegiatan registrasi Client ke dalam sistem untuk memperoleh account.	
<i>Precondition</i>	Seorang Client ingin melakukan register kedalam sistem harus mengakses sistem tersebut.	
<i>Trigger</i>	<i>Usecase</i> ini dilakukan setelah <i>actor</i> mengakses halaman <i>Register</i> .	
<i>Typical Course of</i>	<i>Actor Action</i>	<i>System Response</i>

Events	1: <i>Usecase</i> terinisiasi saat <i>Actor</i> memasukkan data registrasi Client. 5: <i>Usecase berakhir</i> saat <i>actor</i> selesai register.	2: Untuk setiap data yang dimasukan, cek kebenaran pengisian data tersebut. 3: Jika data yang dimasukan sudah benar, simpan data Client kedalam sistem. 4: Sistem mengirim email berisi username dan password kepada Client.
Alternate Courses	3: Jika terdapat kesalahan pengisian data registrasi, maka data tidak dapat tersimpan kedalam database.	
Conclusion	Data Client telah berhasil di registrasi.	
Post Condition	Data Client bertambah kedalam sistem.	

Tabel 4.10 Narasi *usecase Analysis login*

Use Case Name	<i>Login</i>	
Use Case ID	2	
Actor	Client	
Description	<i>Usecase</i> menggambarkan kegiatan <i>login</i> ke dalam sistem dengan memasukkan data pengguna untuk diverifikasi oleh sistem.	
Precondition	Client melakukan <i>login</i> ke sistem harus mengakses terlebih dahulu sistem tersebut.	
Trigger	<i>Usecase</i> ini dilakukan setelah <i>actor</i> mengakses halaman <i>Login</i> .	
Typical Course of Events	<i>Actor Action</i>	<i>System Response</i>
	1: <i>Usecase</i> terinisiasi saat <i>Actor</i> memasukkan data	2: Untuk setiap data pengguna, cek

	verifikasi pengguna yang akan diproses. 4: <i>Usecase berakhir</i> saat <i>actor</i> telah berhasil <i>login</i> ke dalam sistem.	kebenaran pengisian data tersebut. 3: Jika data pengguna terdapat dalam <i>database</i> maka sistem akan menampilkan halaman <i>home</i> dengan tampilan menu sesuai <i>role system</i> pengguna.
Alternate Courses	3: Jika data pengguna tidak terdapat dalam <i>database</i> maka sistem akan mengembalikan pengguna ke halaman <i>home</i> untuk publik.	
Conclusion	<i>Actor</i> berhasil <i>login</i> ke dalam sistem.	
Post Condition	<i>Actor</i> telah berada pada halaman <i>home</i> sesuai <i>role</i> masing-masing.	

Tabel 4.11 Narasi *Usecase Analysis Input Ticket*

Use Case Name	<i>Input Ticket</i>	
Use Case ID	3	
Actor	Client	
Description	<i>Usecase</i> menggambarkan kegiatan memasukkan data <i>Ticket</i> .	
Precondition	Seorang Client yang ingin input <i>ticket</i> harus login terlebih dahulu.	
Trigger	<i>Usecase</i> ini dilakukan setelah <i>login</i> kedalam sistem	
Typical Course of Events	<i>Actor Action</i>	<i>System Response</i>
	1: <i>Usecase</i> terinisiasi saat <i>actor</i> memasukan <i>detail ticket</i> .	2: Untuk setiap data yang di input, cek kebenaran pengguna data tersebut.

	<p>5: <i>Usecase berakhir</i> saat <i>actor</i> telah berhasil membuat <i>ticket</i> baru kedalam sistem.</p>	<p>3: Jika data yang dimasukan telah diisi dengan benar, simpan data <i>ticket</i> kedalam sistem.</p> <p>4. Sistem mengirimkan email yang berisi informasi <i>ticket</i> kepada Client dan manager.</p>
Alternate Courses	3: Jika data yang dimasukan telah diisi dengan benar, simpan data <i>ticket</i> kedalam sistem.	
Conclusion	Data <i>ticket</i> telah berhasil diinput.	
Post Condition	Data <i>ticket</i> sudah bertambah kedalam sistem.	

Tabel 4.12 Narasi *Usecase Analysis View Ticket*

Use Case Name	<i>View Ticket</i>	
Use Case ID	4	
Actor	Client, frondesk, engineer, manager.	
Description	<i>Usecase</i> menggambarkan kegiatan melihat data <i>Ticket</i> .	
Precondition	Seorang <i>Actor</i> yang ingin melihat <i>ticket</i> harus <i>login</i> ke dalam data.	
Trigger	<i>Usecase</i> ini dilakukan setelah <i>login</i> kedalam sistem	
Typical Course of Events	<i>Actor Action</i>	<i>System Response</i>
	1: <i>Usecase</i> terinisiasi saat <i>actor</i> mangakses <i>view ticket</i> .	2: Menampilkan <i>List ticket</i> .

	3: <i>Usecase</i> selesai saat <i>list ticket</i> telah di tampilkan dan dapat dilihat <i>actor</i> .	
Alternate Courses	-	
Conclusion	<i>Actor</i> telah berhasil melihat data <i>ticket</i> .	
Post Condition	<i>List ticket</i> dapat di tampilkan.	

Tabel 4.13 Narasi *Usecase Analysis Assign Ticket*

Use Case Name	<i>Assign Ticket</i>	
Use Case ID	5	
Actor	Frondesk.	
Description	<i>Usecase</i> menggambarkan kegiatan menugaskan teknisi untuk menyelesaikan masalah atau <i>ticket</i> .	
Precondition	Seorang Actor harus <i>login</i> kedalam sistem.	
Trigger	<i>Usecase</i> ini dilakukan setelah <i>login</i> kedalam sistem	
Typical Course of Events	Actor Action	System Response
	1: <i>Usecase</i> terinisiasi saat <i>actor</i> mangakses <i>Assign ticket</i> , dan memilih teknisi yang akan ditugaskan. 5. <i>Usecase</i> berakhir saat <i>actor</i> telah berhasil melakukan <i>Assign Ticket</i> .	2: Sistem melakukan pengecekan terhadap data yang dimasukkan. 3. Jika data yang dimasukkan benar, simpan data kedalam sistem. 4. Sistem mengirimkan email peringatan kepada Client, teknisi, manager.
Alternate Courses	3. Jika data yang dimasukkan salah, tampilkan	

	peringatan.
Conclusion	<i>Actor</i> telah berhasil melakukan <i>Assigned Ticket</i> .
Post Condition	<i>Ticket</i> telah di assign ke salah satu teknisi.

Tabel 4.14 Narasi *Usecase Analysis Input Solusi*

Use Case Name	<i>Input Solusi</i>	
Use Case ID	6	
Actor	Engineer.	
Description	<i>Usecase</i> menggambarkan kegiatan memasukkan data solusi atau penyelesaian terhadap <i>ticket</i> kedalam sistem.	
Precondition	Seorang <i>Actor</i> harus <i>login</i> terlebih dahulu ke dalam system.	
Trigger	<i>Usecase</i> ini dilakukan setelah seluruh kegiatan <i>help desk ticket</i> dimasukan kedalam sistem, dan waktu kegiatan telah berakhir.	
Typical Course of Events	<i>Actor Action</i>	<i>System Response</i>
	1: <i>Usecase</i> terintegrasi saat <i>actor</i> mengakses menu input solusi dan memasukkan data solusi <i>ticket</i> .	2: Sistem melakukan pengecekan terhadap data yang di masukkan. 3: Jika data yang dimasukkan sudah benar, simpan data kedalam sistem.
	5: <i>Usecase</i> berakhir saat <i>actor</i> telah berhasil melakukan input solusi.	4: Sistem mengirimkan email peringatan kepada <i>Client</i> , teknisi dan manager.

Alternate Courses	3. Jika data yang dimasukkan salah, tampilkan peringatan.	
Conclusion	Actor telah berhasil melakukan <i>Input solusi</i> .	
Post Condition	Solusi ticket telah diinput kedalam sistem.	

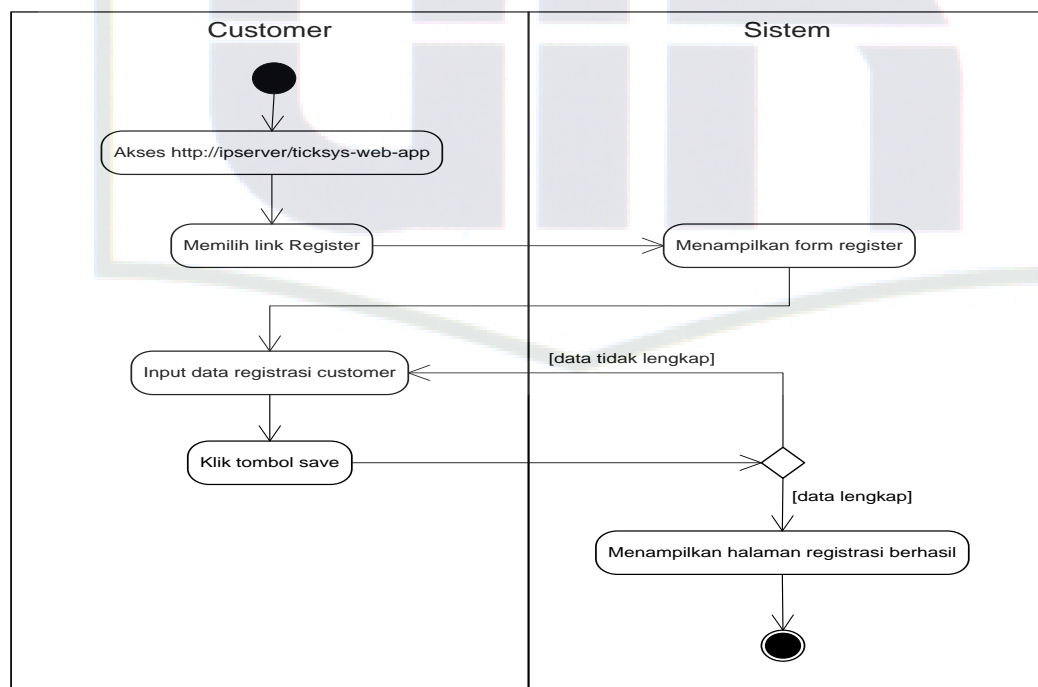
Tabel 4.15 Narasi *Usecase Analysis* Lihat Laporan

Use Case Name	Lihat Laporan	
Use Case ID	7	
Actor	Manager.	
Description	<i>Usecase</i> menggambarkan kegiatan melihat laporan <i>ticket</i> .	
Precondition	Seorang Actor harus <i>login</i> terlebih dahulu ke dalam sistem.	
Trigger	<i>Usecase</i> ini dilakukan setelah seluruh kegiatan <i>help desk ticket system</i> selesai dikerjakan kemudian dimasukan kedalam sistem, kemudian buat laporan untuk manager.	
Typical Course of Events	<i>Actor Action</i>	<i>System Response</i>
	1: Klik menu " <i>ticket</i> ". 3: Klik menu " <i>Buat Laporan Ticket</i> ".	2: Menampilkan halaman <i>Ticket</i> . 4: Menampilkan halaman Laporan <i>Ticket</i> . 6: Jika laporan data nya sudah di cek kebenarannya, kemudian laporan diberikan kepada manager.
Alternate Courses	6: Jika laporan data nya sudah di cek kebenarannya, kemudian laporan diberikan kepada manager.	

Conclusion	Actor telah berhasil membuat Laporan <i>ticket</i> .
Post Condition	Laporan <i>ticket</i> telah berhasil dilakukan kemudian diberikan kepada manager.

4.2.7.3 Pemodelan Activity Diagram

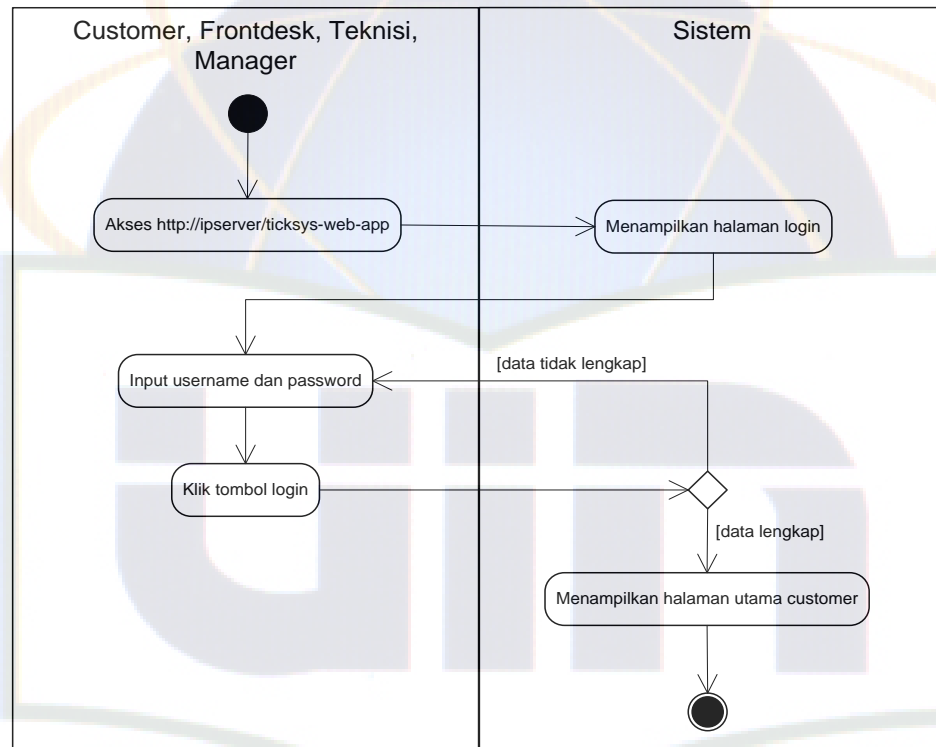
Gambar 4.5 menggambarkan *activity diagram* yang memperlihatkan alur jalannya *Helpdesk Ticketing System* dan memperlihatkan siapa saja yang bertanggung jawab terhadap setiap tahap yang dilakukan.



Gambar 4.5 Activity Diagram Client

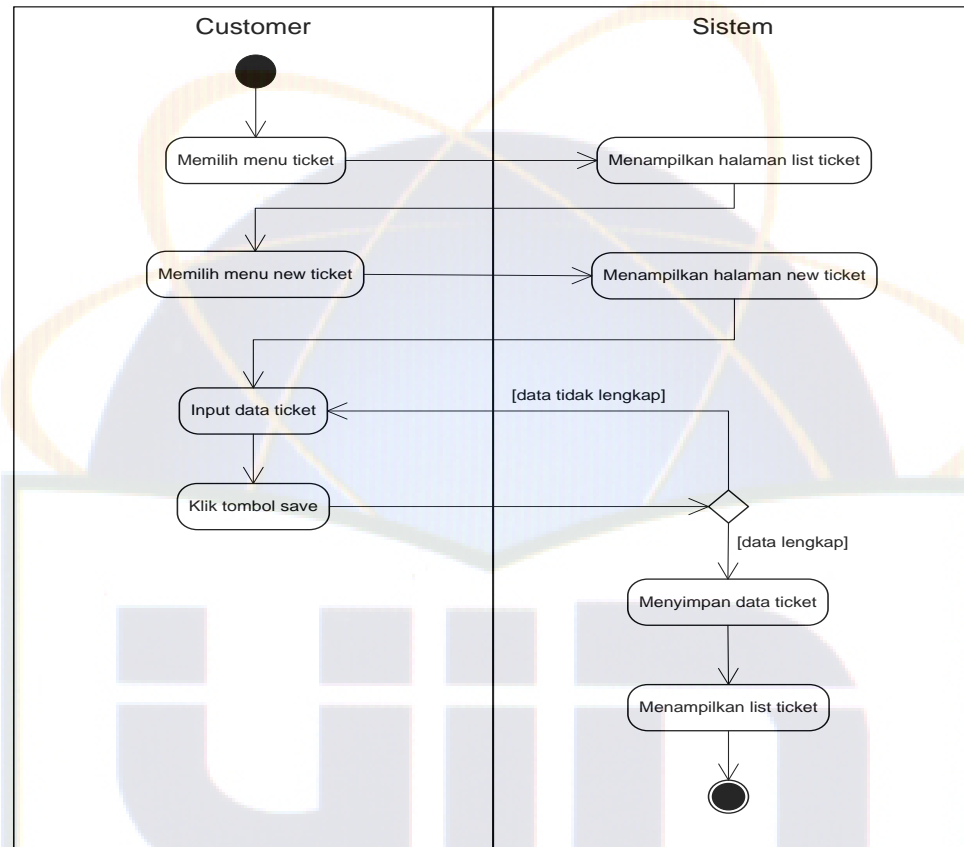
Alur jalannya sistem yang diperlihatkan pada *activity diagram Client* dimulai saat dilakukannya proses pengaksesan website setelah itu melakukan proses *Register* yang dapat dilakukan oleh *Client*, dan muncul tampilan form register. Kemudian *Client* melakukan input data *Client*, jika data valid maka

proses akan berhasil akan tetapi jika data tidak valid maka proses akan kembali ke awal, kemudian data disimpan ke dalam sistem, setelah data semua lengkap dapat menampilkan halaman register berhasil. Setelah itu proses selesai.



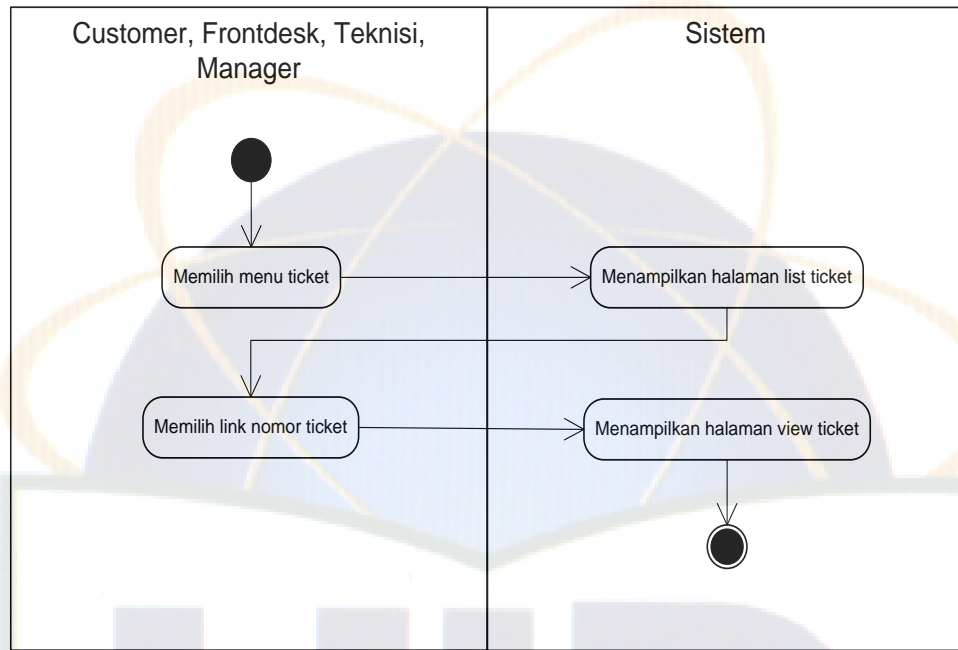
Gambar 4.6 Activity Diagram Login

Alur jalannya sistem yang diperlihatkan adalah *Client*, *frontdesk*, *teknisi*, *manager* harus membuka situs website tersebut kemudian melakukan login terlebih dahulu, jika username dan password valid maka proses akan berhasil jika data tidak valid maka akan kembali ke menu awal, jika data sudah lengkap semua maka dapat menampilkan halaman utama *Client*, setelah itu proses selesai.



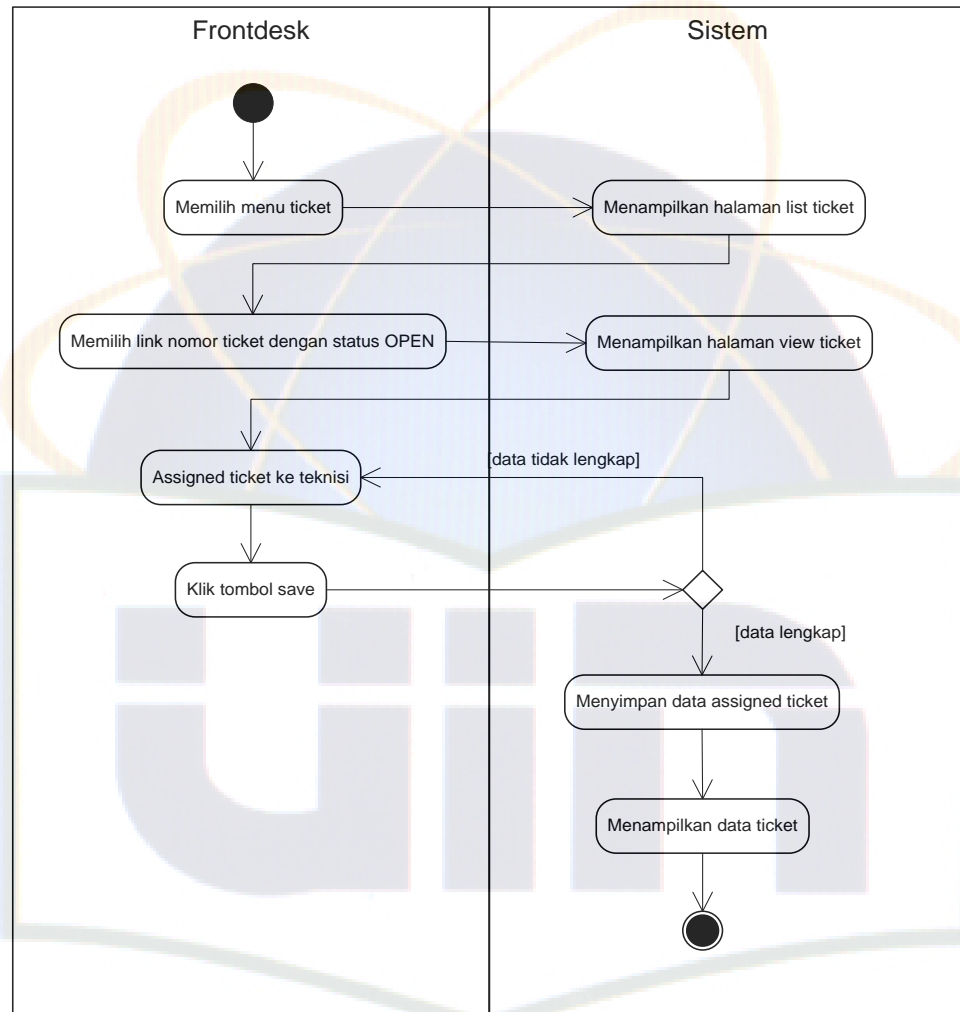
Gambar 4.7 *Activity Diagram Input Ticket*

Alur jalannya sistem yang diperlihatkan adalah Client memilih menu ticket dan halaman ticket dapat ditampilkan, kemudian memilih new ticket dan dapat menampilkan halaman new ticket, setelah itu ticket diinput jika data benar maka data dapat disimpan kedalam sistem, jika data salah maka tidak dapat diproses, jika data benar maka si simpan kedalam sistem dan dapat menampilkan list ticket, setelah itu proses selesai.



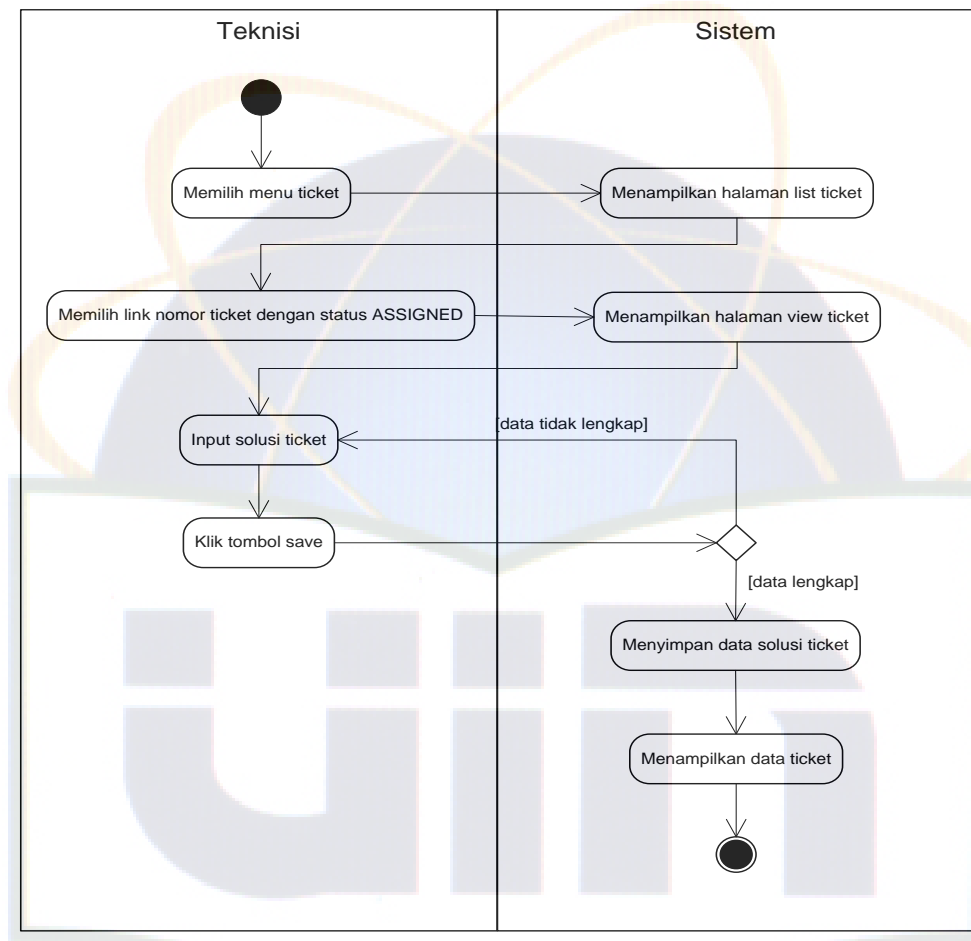
Gambar 4.8 *Activity Diagram view ticket*

Alur jalannya sistem yang diperlihatkan adalah *Client, frontdesk, teknisi, manager* memilih menu ticket, kemudian dapat menampilkan halaman list ticket, kemudian memilih link nomor ticket, dan dapat menampilkan halaman view ticket, setelah itu proses selesai.



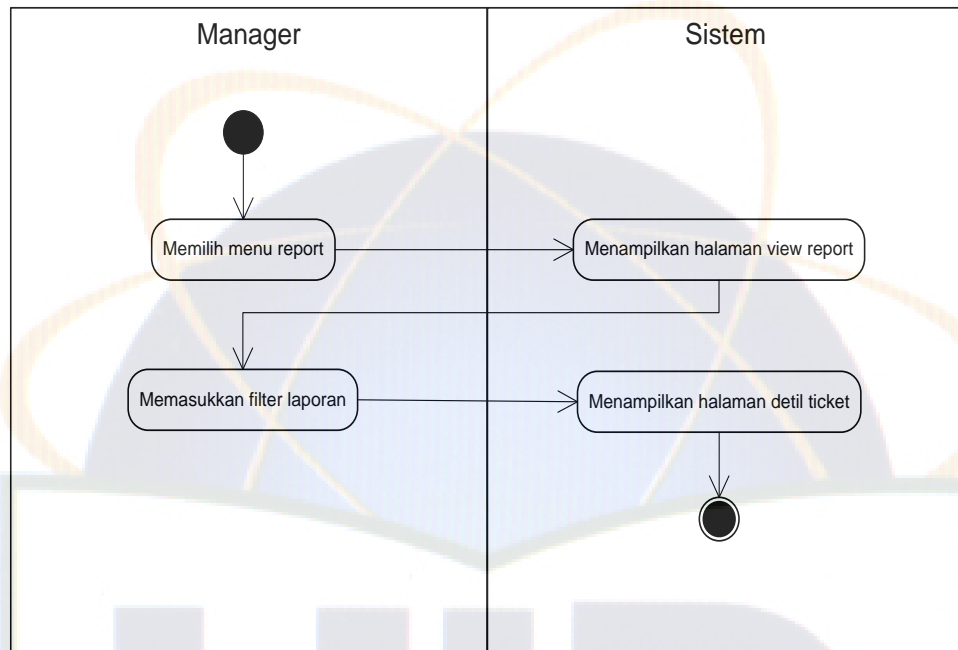
Gambar 4.8 Activity Diagram Assigned ticket

Alur jalan proses nya adalah *frontdesk* memilih menu *ticket* dan dapat menampilkan halaman *list ticket*, setelah itu memilih link nomor *ticket* dengan status OPEN dan dapat menampilkan halaman *view ticket*, kemudian melakukan proses *assigned ticket* ke teknisi, jika data tidak lengkap maka data tidak dapat diproses, tapi jika data lengkap maka data dapat diproses dan dapat disimpan kedalam sistem, kemudian data juga disimpan ke *assigned ticket* dan dapat menampilkan data *ticket*, setelah itu proses berakhir.



Gambar 4.9 *Activity Diagram Input solusi*

Alur jalan proses nya adalah teknisi memilih menu ticket dan dapat menampilkan halaman list ticket, kemudian memilih link nomor ticket dengan status ASSIGNED dan dapat menampilkan halaman view ticket, kemudian input solusi ticket, jika data salah maka tidak dapat diproses, tapi jika data benar maka data dapat diproses dan data disimpan, kemudian menyimpan data solusi ticket dan dapat menampilkan data ticket, setelah itu proses berakhir.



Gambar 4.10 *Activity Diagram View Laporan*

Alur jalan proses nya adalah manager memilih menu report, dan dapat menampilkan halaman view report, kemudian memasukkan filter laporan, dan dapat menampilkan halaman detail ticket, setelah itu proses berakhir.

4.3 *System Design (Sistem Desain)*

4.3.1 *Desain dan Pemodelan Berorientasi Obyek*

4.3.1.1 *Narasi Usecase Design*

Narasi *usecase design* merupakan pemaparan naratif penjelasan tentang kegiatan yang dilakukan oleh *actor* dan respon yang diberikan oleh sistem sesuai dengan yang terjadi pada perangkat lunak *Helpdesk ticketing system*.

Tabel 4.16 – Tabel 4.22 adalah narasi *usecase design* dari *Helpdesk ticketing system*.

Tabel 4.16 Narasi *Usecase Design Register*

Use Case Name	<i>Register</i>	
Use Case ID	1	
Actor	Client	
Description	<i>Usecase</i> menggambarkan kegiatan registrasi <i>Client</i> ke dalam sistem untuk memperoleh account.	
Precondition	Seorang <i>Client</i> ingin melakukan register kedalam sistem harus mengakses sistem tersebut.	
Trigger	<i>Usecase</i> ini dilakukan setelah <i>actor</i> membuka url <i>http://ipserver/ticksys-wb-app</i> pada browser.	
Typical Course of Events	<i>Actor Action</i>	<i>System Response</i>
	1: Klik menu “ <i>Register</i> ”. 3: Pilih nama organisasi, input Client name, input email, input address, input office phone, input mobile phone, input verifikasi. 4: Klik tombol “ <i>Save</i> ”.	2: Menampilkan halaman <i>Register</i> . 5: cek data registrasi. 6: Jika seluruh data sudah benar, simpan kedalam ticksys_Client. 7: sistem mengirimkan email dan username dan password kepada Client. 8: Menampilkan halaman informasi registrasi berhasil.
Alternate Courses	4: Klik tombol “back”. 5: Menampilkan halaman login.	
Conclusion	<i>Actor</i> berhasil <i>Registrasi</i> ke dalam sistem.	

Post Condition	Actor berhasil registrasi.
-----------------------	----------------------------

Tabel 4.17 Narasi Usecase Design Login

Use Case Name	Login	
Use Case ID	2	
Actor	Client,frondesk, teknisi, manager	
Description	Usecase menggambarkan kegiatan <i>login</i> ke dalam sistem dengan memasukkan data pengguna untuk diverifikasi oleh sistem.	
Precondition	Seorang Client ingin melakukan <i>login</i> ke sistem harus mengakses terlebih dahulu sistem tersebut.	
Trigger	Usecase ini dilakukan setelah <i>actor</i> membuka url <i>http://ipserver/ticksys-wb-app</i> pada <i>browser</i> .	
Typical Course of Events	Actor Action	System Response
	1: Klik menu “Login”. 3: Input <i>username</i> , <i>password</i> . 4: Klik tombol “Login”.	2: Menampilkan halaman <i>Login</i> . 5: Sistem mengecek apakah <i>username</i> , <i>password</i> . telah dimasukkan. 6: Verifikasi data pengguna dan cek <i>role</i> pengguna ke tabel <i>sys_user</i> . 7: Jika data pengguna terdapat dalam <i>database</i> maka sistem akan menampilkan <i>Alert</i> :

	8: Klik tombol "OK".	Login Berhasil! 9: Menampilkan halaman <i>home</i> dengan tampilan menu sesuai <i>role</i> pengguna.
Alternate Courses	4: Klik tombol "Reset". 5: Sistem menghapus nama pengguna dan <i>password</i> yang telah diketikkan di <i>form Login</i> . 7: Jika nama pengguna dan <i>password</i> tidak terdapat dalam <i>database</i> , sistem menampilkan <i>alert: Username dan/atau Password Salah</i> .	
Conclusion	Actor berhasil <i>login</i> ke dalam sistem.	
Post Condition	Actor telah berada di halaman <i>home</i> .	

Tabel 4.18 Narasi Usecase Design Input Ticket

Use Case Name	<i>Input Ticket</i>	
Use Case ID	3	
Actor	<i>Client</i>	
Description	Usecase menggambarkan kegiatan memasukkan data <i>ticket</i> .	
Precondition	Seorang Client ingin <i>input ticket</i> harus login terlebih dahulu.	
Trigger	Usecase ini dilakukan setelah <i>login</i> kedalam sistem.	
Typical Course of Events	Actor Action	System Response
	1: Usecase terinisiasi saat actor memasukkan <i>detail ticket</i> .	2: untuk setiap data yang di input, di cek kebenaran pengguna data tersebut. 3. Jika data yang dimasukkan telah diisi dengan benar, simpan

	<p>5: <i>Usecase</i> berakhir saat <i>actor</i> telah berhasil memuat <i>ticket</i> baru kedalam sistem.</p>	<p>data <i>ticket</i> kedalam sistem.</p> <p>4. Sistem mengirimkan email yang berisi informasi <i>ticket</i> kepada <i>Client</i> dan <i>manager</i>.</p>
Alternate Courses	3: Jika data yang dimasukkan telah diisi dengan benar, simpan data <i>ticket</i> kedalam sistem.	
Conclusion	Data <i>ticket</i> telah berhasil di input.	
Post Condition	Data <i>ticket</i> sudah bertambah kedalam sistem.	

Tabel 4.19 Narasi *Usecase Design View Ticket*

Use Case Name	View Ticket	
Use Case ID	4	
Actor	<i>Client, frondesk, teknisi, manager.</i>	
Description	<i>Usecase</i> menggambarkan kegiatan melihat data <i>Ticket</i> .	
Precondition	Seorang Actor yang ingin melihat <i>ticket</i> harus <i>login</i> ke dalam data.	
Trigger	<i>Usecase</i> ini dilakukan setelah <i>login</i> kedalam sistem	
Typical Course of Events	Actor Action	System Response
	<p>1: <i>Usecase</i> terinisiasi saat <i>actor</i> mangakses view <i>ticket</i>.</p> <p>3: <i>Usecase</i> selesai saat <i>list ticket</i> telah di tampilkan dan dapat</p>	2: Untuk menampilkan <i>List ticket</i> .

	dilihat actor.	
Alternate Courses	-	
Conclusion	Actor telah berhasil melihat data <i>ticket</i> .	
Post Condition	<i>List ticket</i> dapat di tampilkan.	

Tabel 4.20 Narasi *Usecase Design Assign Ticket*

Use Case Name	<i>Assign Ticket</i>	
Use Case ID	5	
Actor	<i>Frondesk.</i>	
Description	<i>Usecase</i> menggambarkan kegiatan menugaskan teknisi untuk menyelesaikan masalah atau <i>ticket</i> .	
Precondition	Seorang Actor harus <i>login</i> kedalam sistem.	
Trigger	<i>Usecase</i> ini dilakukan setelah <i>login</i> kedalam sistem	
Typical Course of Events	Actor Action	System Response
	1: <i>Usecase</i> terinisiasi saat actor mangakses <i>Assign ticket</i> , dan memilih teknisi yang akan ditugaskan.	2: Sistem melakukan pengecekan terhadap data yang dimasukkan. 3. Jika data yang dimasukkan benar, simpan data kedalam sistem. 4. Sistem mengirimkan email peringatan kepada Client, teknisi, manager.
	5. <i>Usecase</i> berakhir saat actor telah berhasil melakukan <i>Assign Ticket</i> .	
Alternate Courses	3. Jika data yang dimasukkan salah, tampilkan peringatan.	
Conclusion	Actor telah berhasil melakukan <i>Assigned Ticket</i> .	
Post Condition	<i>Ticket</i> telah di assign ke salah satu engineer.	

Tabel 4.21 Narasi *Usecase Design Input Solusi*

<i>Use Case Name</i>	<i>Input Solusi</i>	
<i>Use Case ID</i>	6	
<i>Actor</i>	Teknisi	
<i>Description</i>	<i>Usecase</i> menggambarkan kegiatan memasukkan data solusi atau penyelesaian terhadap <i>ticket</i> kedalam sistem.	
<i>Precondition</i>	Seorang Actor harus <i>login</i> terlebih dahulu ke dalam system.	
<i>Trigger</i>	<i>Usecase</i> ini dilakukan setelah seluruh kegiatan <i>help desk ticket</i> dimasukan kedalam sistem, dan waktu kegiatan telah berakhir.	
<i>Typical Course of Events</i>	<i>Actor Action</i>	<i>System Response</i>
	<p>1: <i>Usecase</i> terintegrasi saat actor mengakses menu input solusi dan memasukkan data solusi ticket.</p> <p>5: <i>Usecase</i> berakhir saat actor telah berhasil melakukan input solusi.</p>	<p>2: Sistem melakukan pengecekan terhadap data yang di masukkan.</p> <p>3: Jika data yang dimasukkan sudah benar, simpan data kedalam sistem.</p> <p>4: Sistem mengirimkan email peringatan kepada <i>Client</i>, teknisi dan manager.</p>
<i>Alternate Courses</i>	3. Jika data yang dimasukkan salah, tampilkan peringatan.	

Conclusion	Actor telah berhasil melakukan <i>Input solusi</i> .
Post Condition	Solusi ticket telah diinput kedalam sistem.

Tabel 4.22 Narasi *Usecase Design* Lihat Laporan

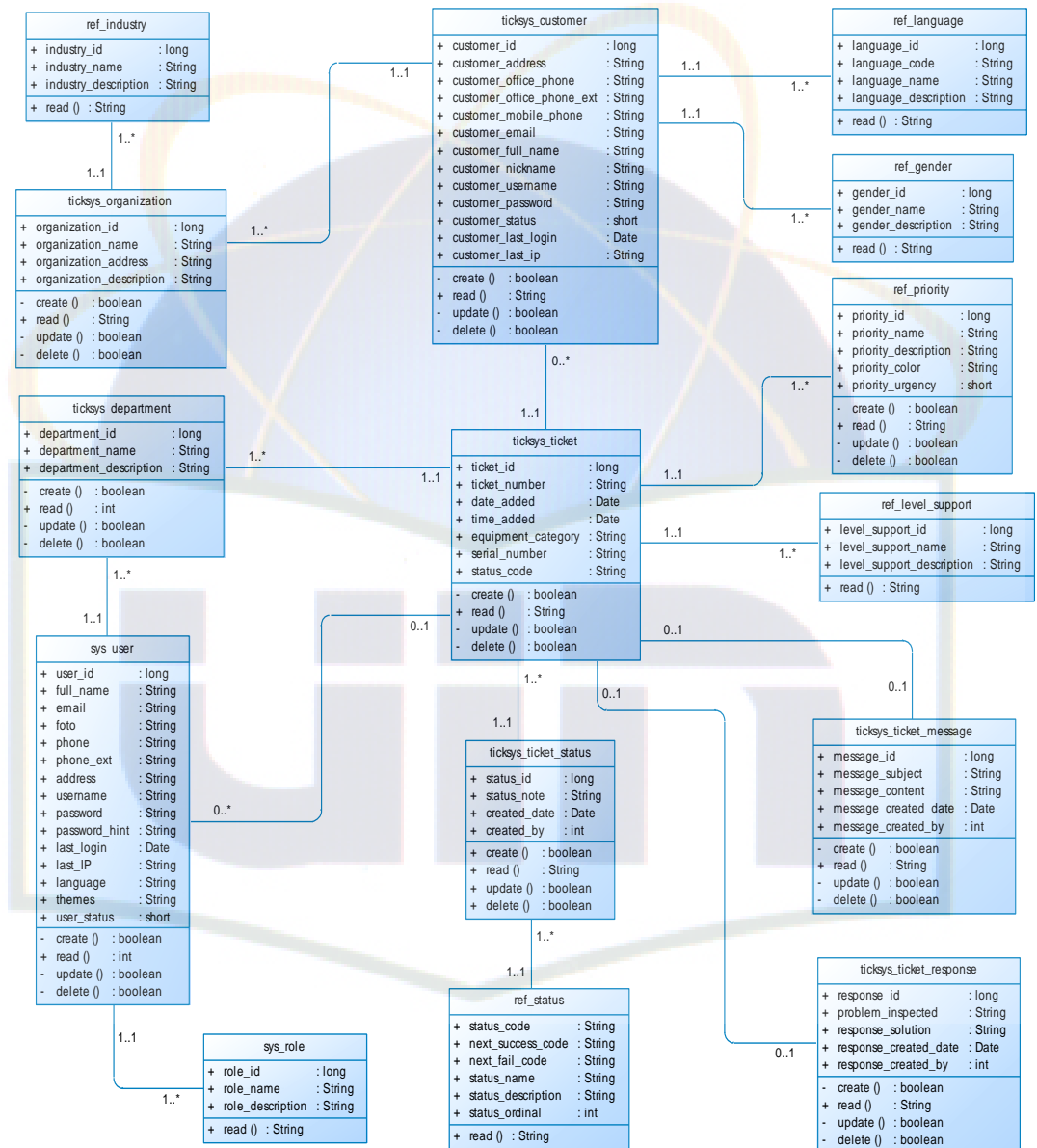
Use Case Name	Lihat Laporan	
Use Case ID	7	
Actor	Manager.	
Description	<i>Usecase</i> menggambarkan kegiatan melihat laporan <i>ticket</i> .	
Precondition	Seorang Actor harus <i>login</i> terlebih dahulu ke dalam sistem.	
Trigger	<i>Usecase</i> ini dilakukan setelah seluruh kegiatan <i>helpdesk ticket system</i> selesai di kerjakan kemudian dimasukan kedalam sistem, kemudian buat laporan untuk manager.	
Typical Course of Events	<i>Actor Action</i>	<i>System Response</i>
	1: Klik menu " <i>ticket</i> ". 3: Klik menu " <i>Buat Laporan Ticket</i> ".	2: Menampilkan halaman <i>Ticket</i> . 4: Menampilkan halaman Laporan <i>Ticket</i> . 6: Jika laporan data nya sudah di cek kebenarannya, kemudian laporan diberikan kepada manager.
Alternate Courses	6: Jika laporan data nya sudah di cek kebenarannya, kemudian laporan diberikan kepada manager.	
Conclusion	Actor telah berhasil membuat Laporan <i>ticket</i> .	
Post Condition	Laporan <i>ticket</i> telah berhasil dilakukan kemudian	

	diberikan kepada manager.
--	---------------------------

4.3.1.2 Pemodelan *Class Diagram*

Visualisasi dari struktur *object* sistem yang diajukan, digambarkan dalam *class diagram*. Adapun langkah-langkah untuk menentukan *object* yang terlibat adalah sebagai berikut adalah:

1. Client menjadi ticksys_Client.
2. Frondesk menjadi sys_user.
3. Role pengguna menjadi sys_role.
4. Organisasi menjadi ticksys_organization.
5. Industri menjadi ref_industry
6. Departement menjadi ticksys_departement
7. Language manjadi ref_language
8. Gender menjadi ref_gender
9. Priority menjadi ref_priority
10. Level support menjadi ref_level_support
11. Ticket status menjadi ticksys_ticket_status
12. Ticket message menjadi ticksys_ticket_message
13. Ticket response menjadi ticksys_ticket_response
14. Status menjadi ref_status
15. Ticket menjadi ticksys_ticket

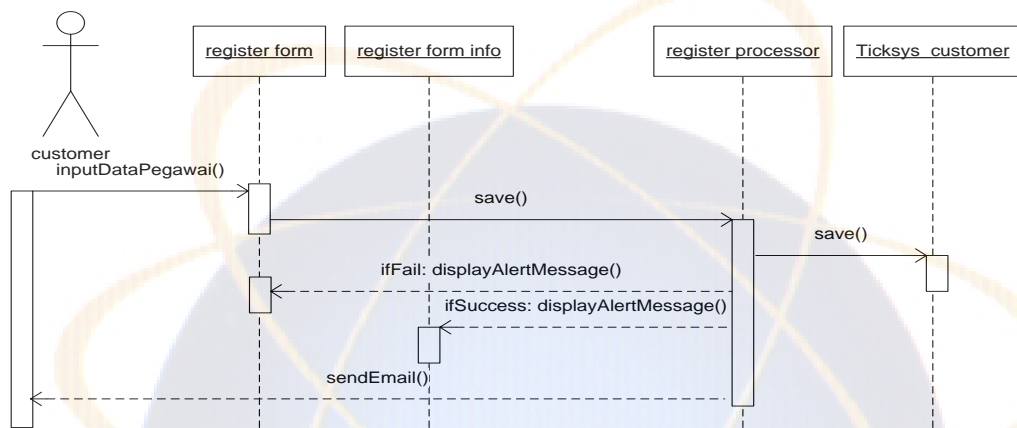


Gambar 4.11 *Class Diagram Helpdesk Ticketing system*

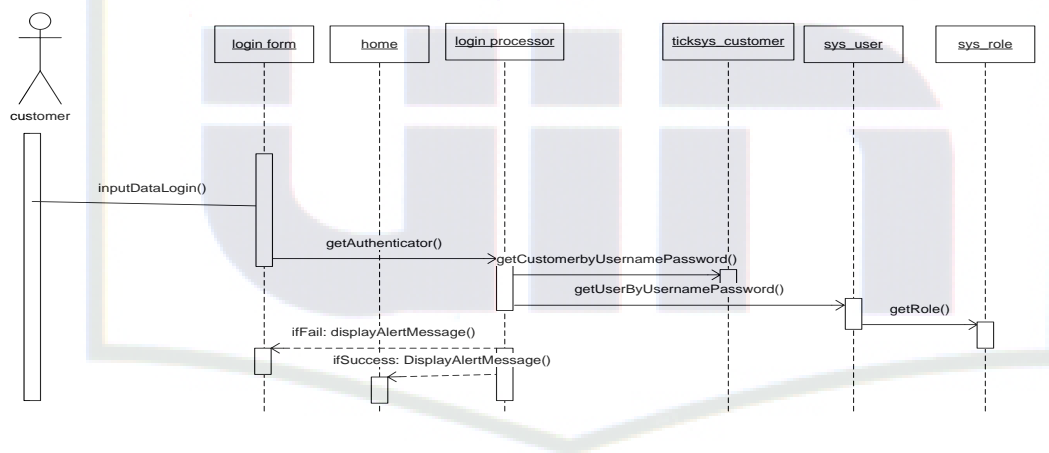
4.3.1.3 Pemodelan *Sequence Diagram*

Sequence diagram digunakan untuk memperlihatkan aliran fungsionalitas dan kontrol yang melalui tiap obyek pada *helpdesk ticketing system*.

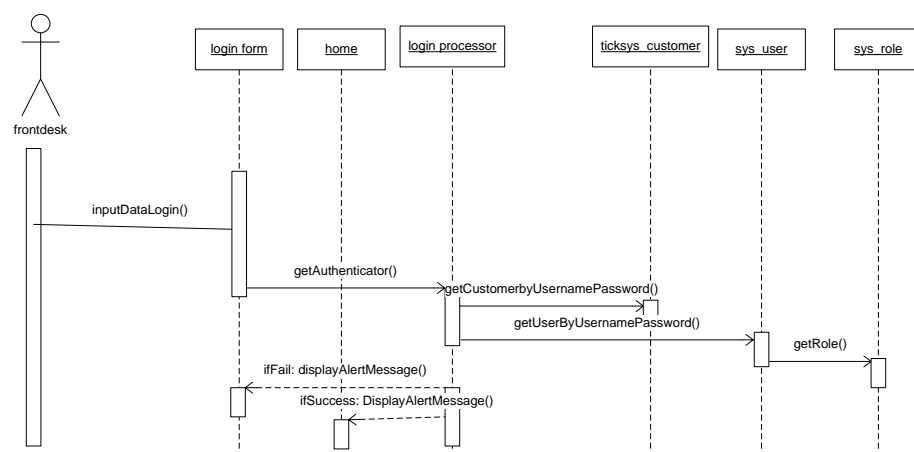
Gambar 4.12 – Gambar 4.29 adalah gambaran *sequence diagram helpdesk ticketing system*.



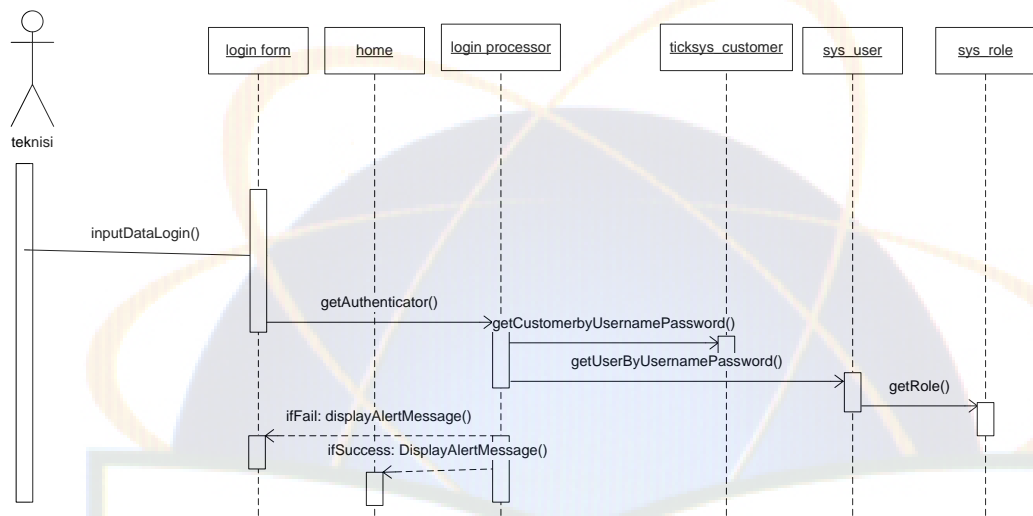
Gambar 4.12 Sequence Diagram Register Helpdesk ticketing system



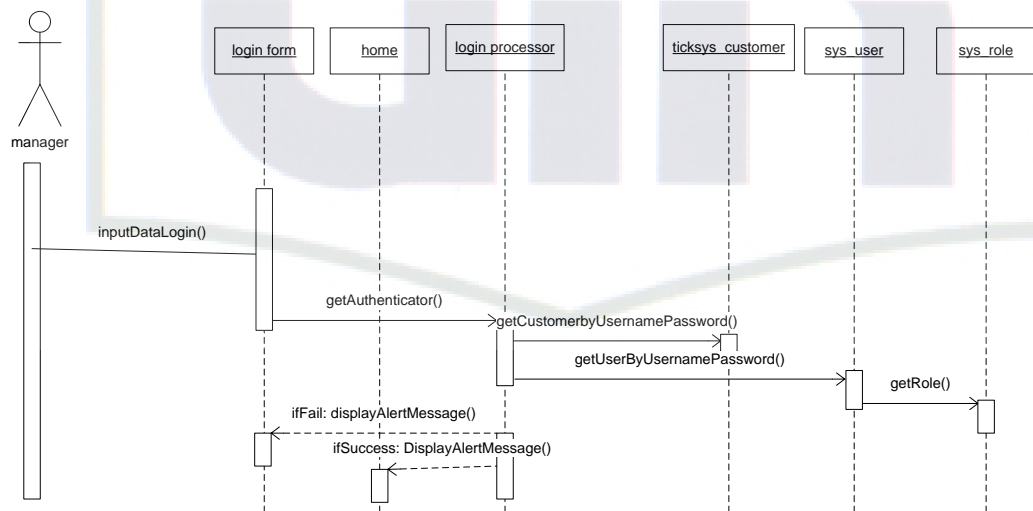
Gambar 4.13 Sequence Diagram Login Client Helpdesk Ticketing System



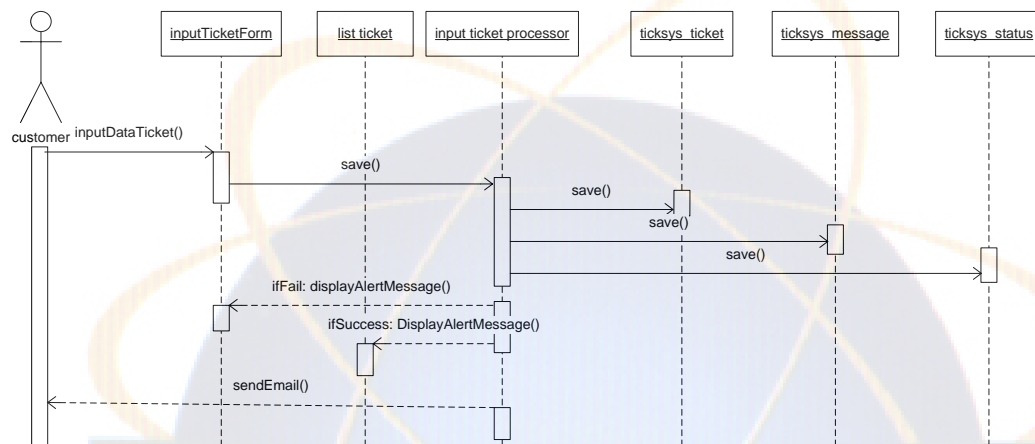
Gambar 4.14 Sequence Diagram Login Frontdesk Helpdesk ticketingSystem



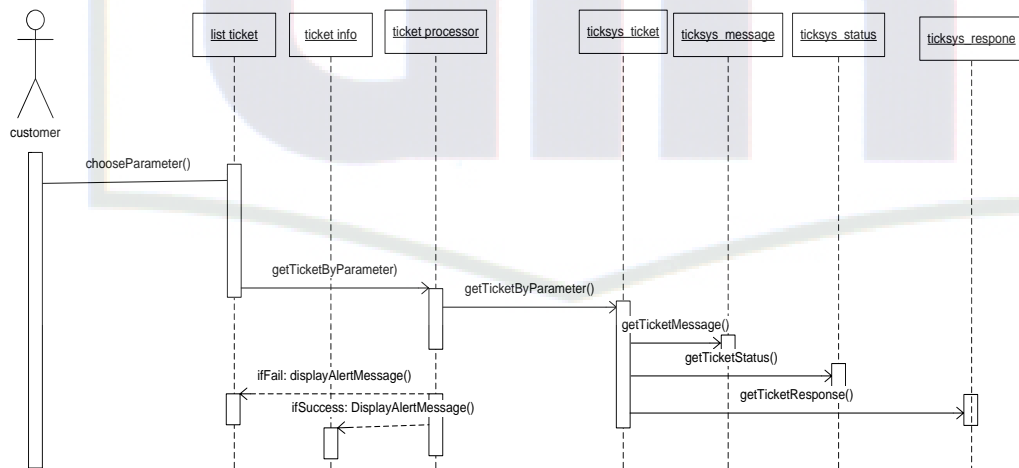
Gambar 4.15 *Sequence Diagram Login Teknisi Helpdesk ticketingSystem*



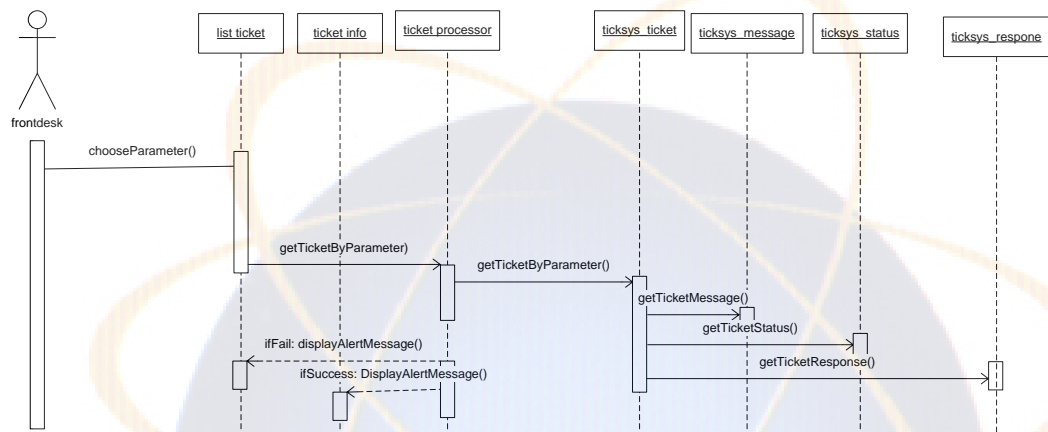
Gambar 4.16 *Sequence Diagram Login manager Helpdesk system*



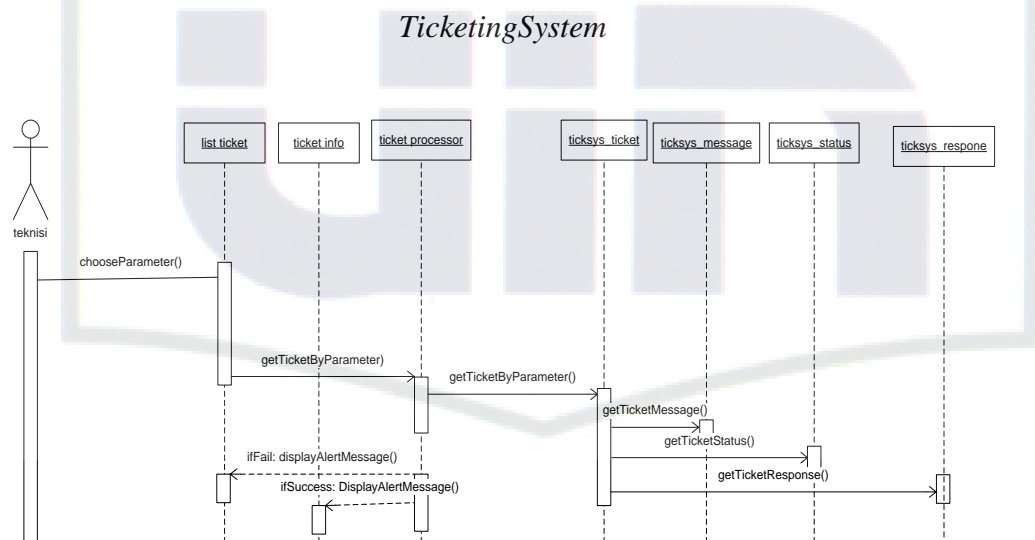
Gambar 4.17 *Sequence Diagram Input Ticket Helpdesk ticketing system*



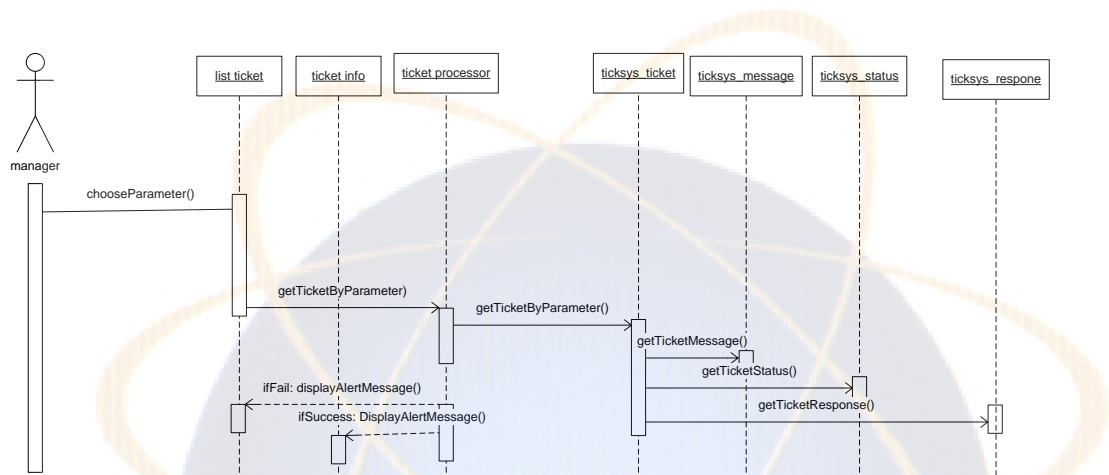
Gambar 4.18 *Sequence Diagram View Ticket Client Helpdesk Ticketing System*



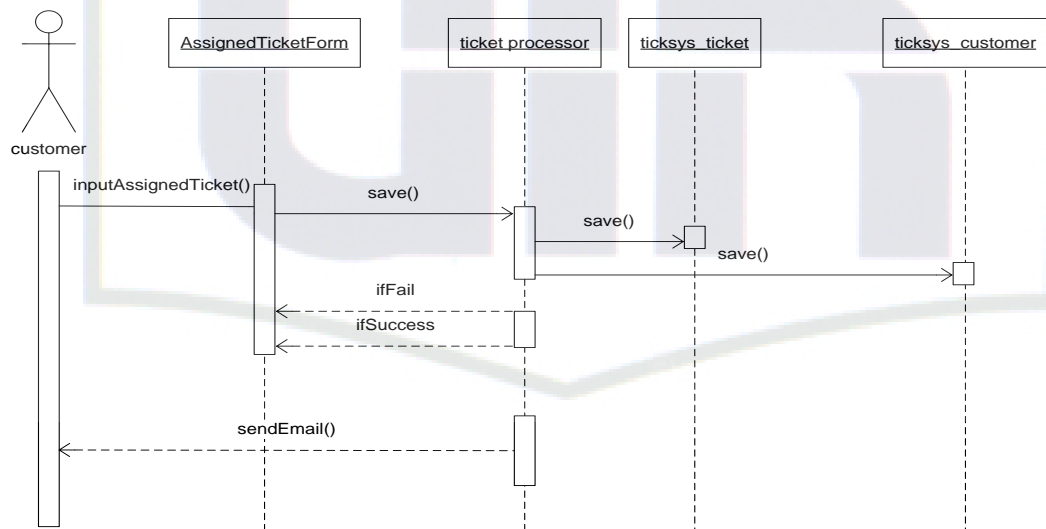
Gambar 4.19 *Sequence Diagram View Ticket Frontdesk Helpdesk*



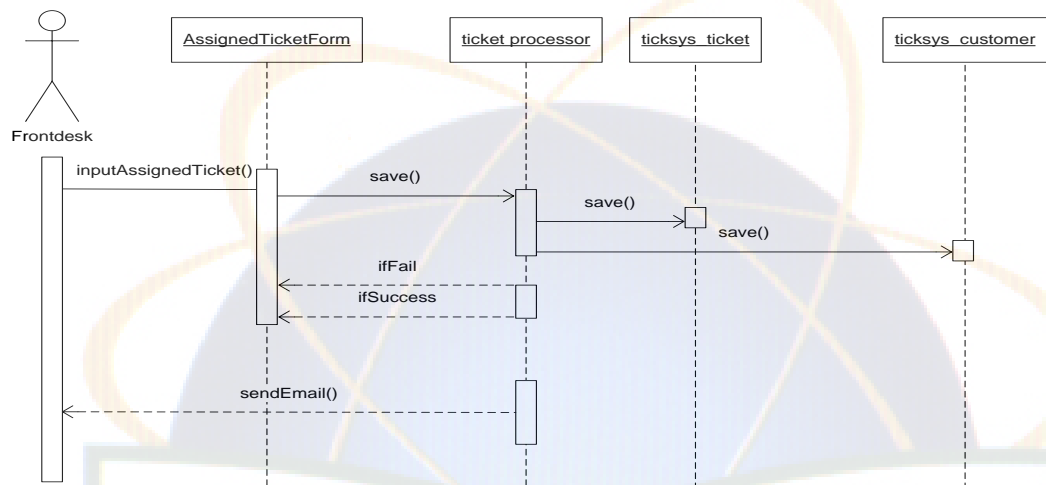
Gambar 4.20 *Sequence Diagram View Ticket Teknisi Helpdesk TicketingSystem*



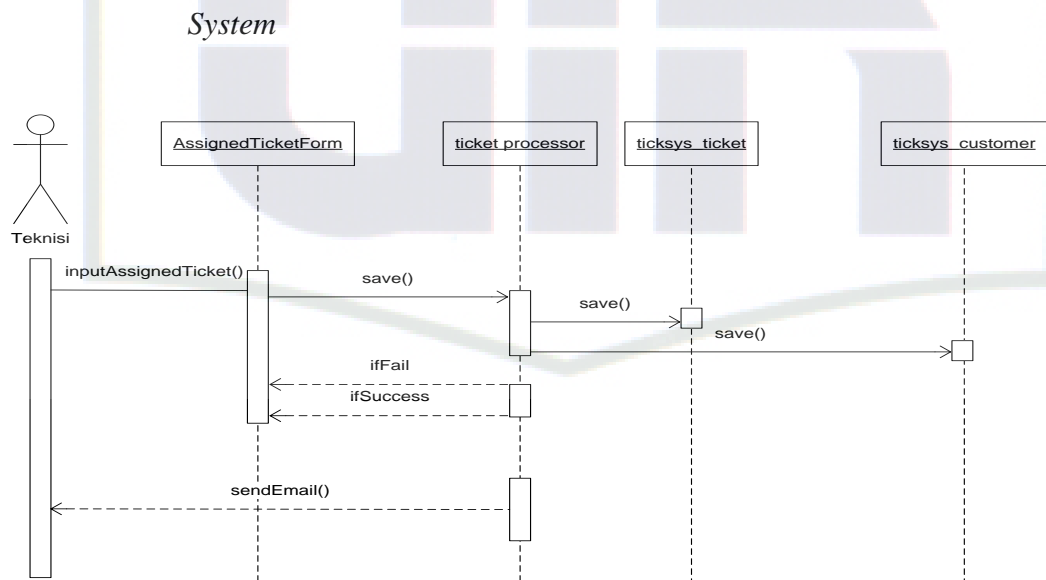
Gambar 4.21 Sequence Diagram View Ticket manager Helpdesk TicketingSystem



Gambar 4.22 Sequence Diagram Assigned Ticket ClientHelpdesk Ticketing System

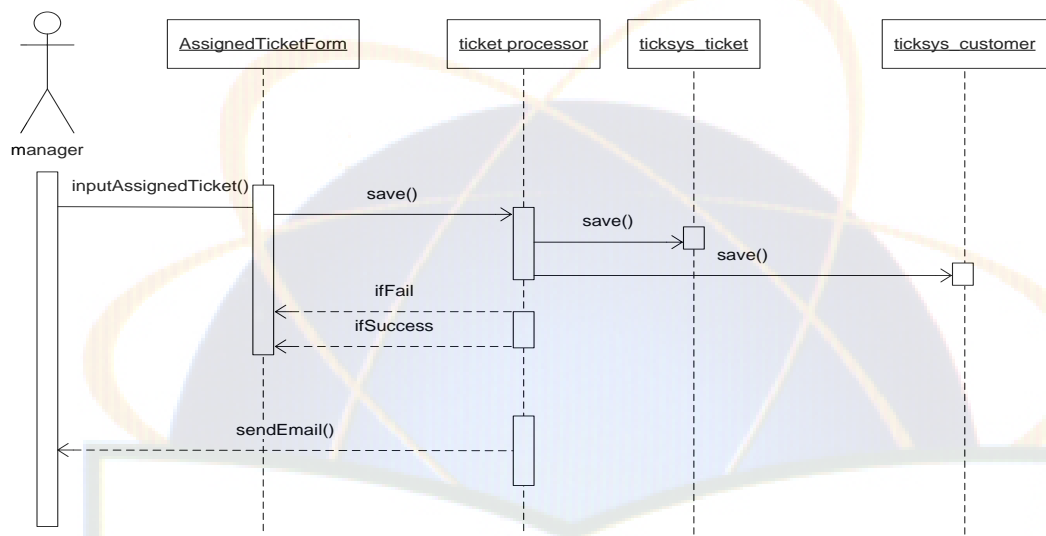


Gambar 4.23 *Sequence Diagram Assigned Ticket Frontdesk Helpdesk Ticketing*

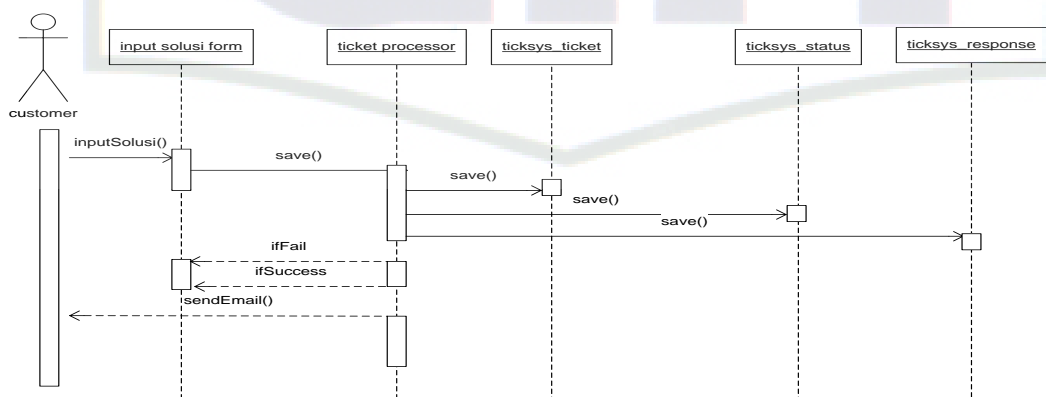


Gambar 4.24 *Sequence Diagram Assigned Ticket teknisi Helpdesk Ticketing*

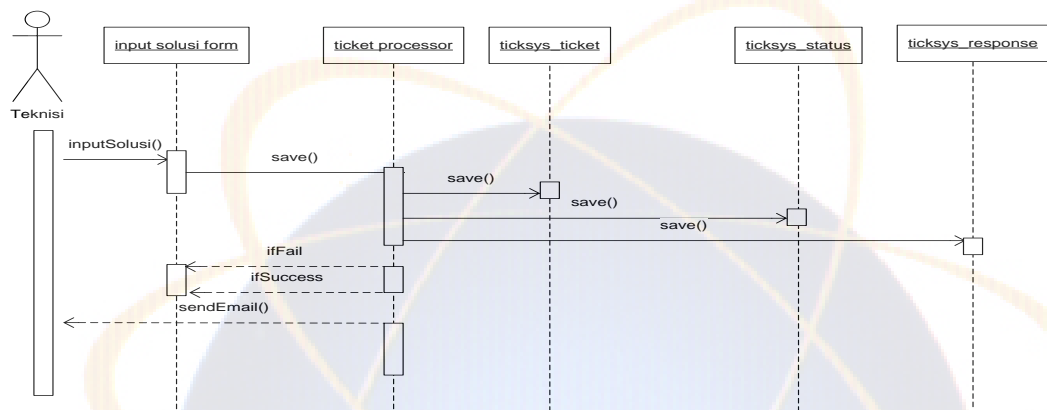
System



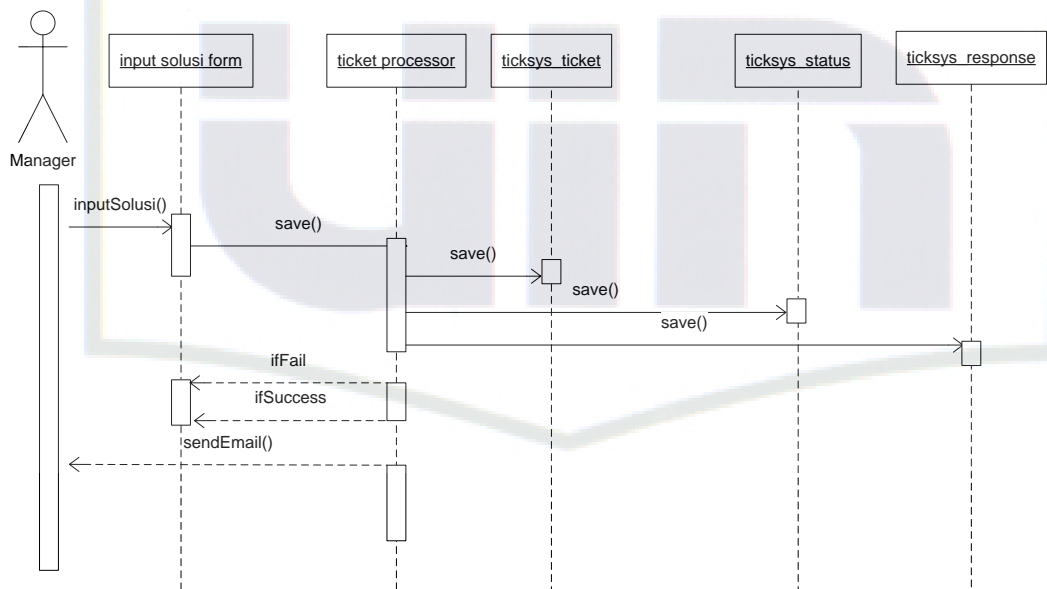
Gambar 4.25 *Sequence Diagram Assigned Ticket manager Helpdesk Ticketing System*



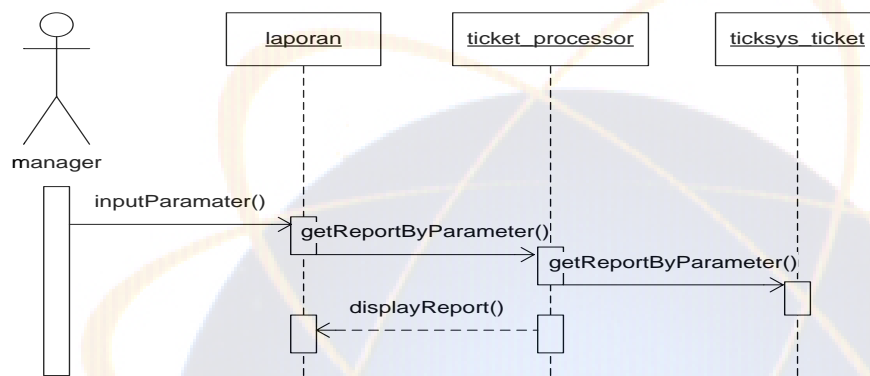
Gambar 4.26 *Sequence Diagram Input Solusi Client Helpdesk Ticketing System*



Gambar 4.27 Sequence Diagram Input Solusi Teknisi Helpdesk Ticketing System



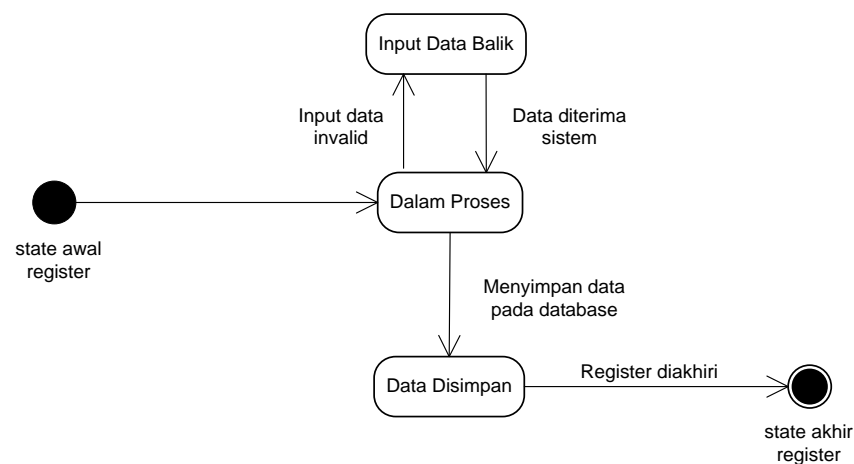
Gambar 4.28 Sequence Diagram Input solusi manager Helpdesk Ticketing System



Gambar 4.29 *Sequence Diagram* Lihat Laporan Helpdesk Ticketing System

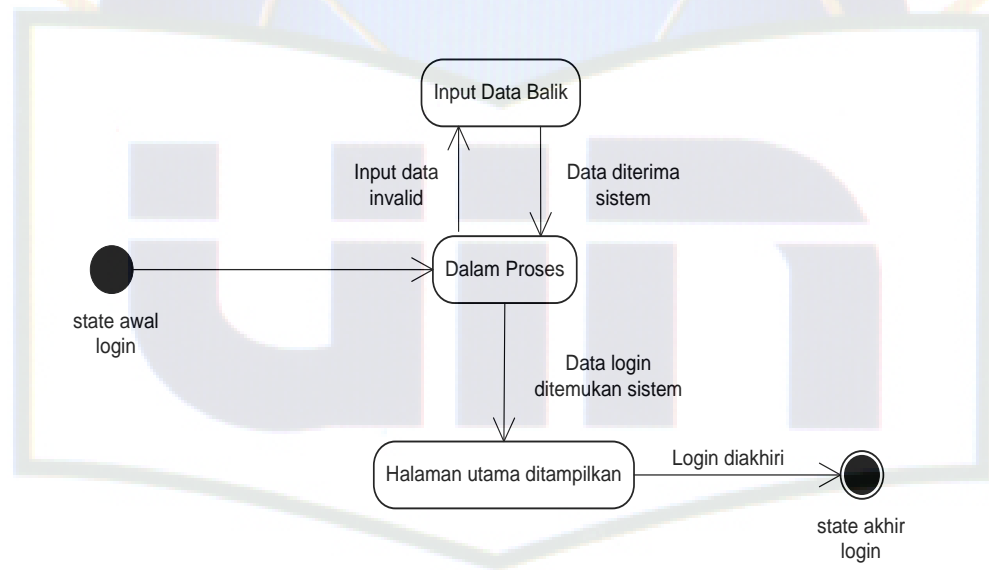
4.3.1.4 Pemodelan Statechart Diagram

Statechart diagram memperlihatkan siklus hidup suatu obyek pada *Helpdesk ticketing system*, sejak obyek tersebut mulai dibuat hingga obyek tersebut menghilang. Namun, tidak semua obyek yang terdapat pada sistem akan dibuat *state diagram*-nya, hanya obyek yang memiliki perubahan status yang akan dibuat *state*-nya.



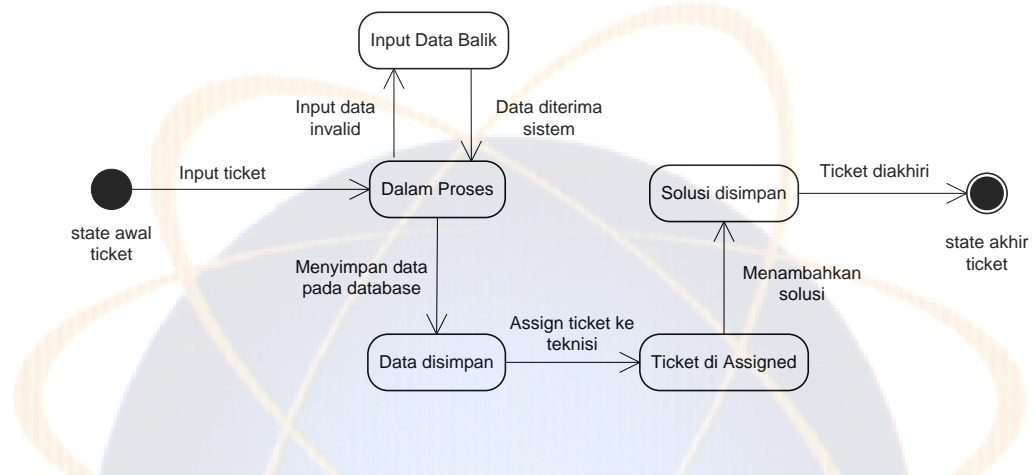
Gambar 4.30 *State Diagram* Register Client

State diagram register Client tersebut proses nya adalah Client melakukan registrasi terlebih dahulu, kemudian data di proses jika data benar maka akan masuk kedalam sistem, tapi jika data salah maka akan kembali ke menu awal, kemudian data yang benar di simpan kedalam database, setelah itu proses registrasi selesai.



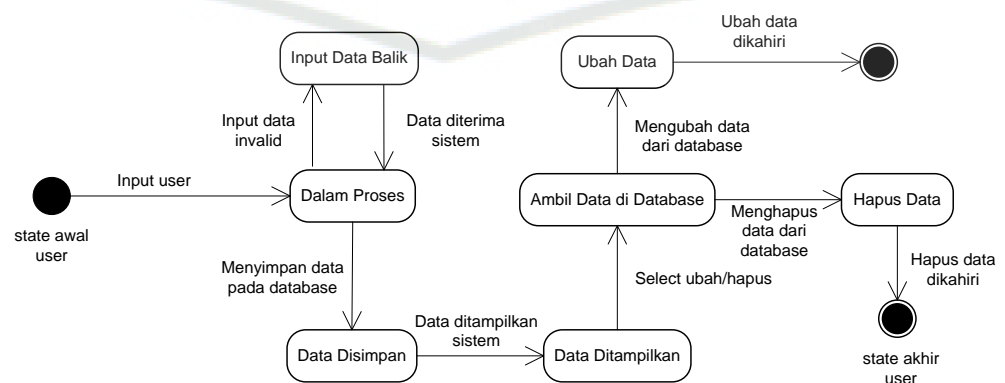
Gambar 4.31 *State Diagram Login*

Alur jalannya sistem adalah Client melakukan login terlebih dahulu, kemudia data diproses, jika data benar maka data bisa di terima kedalam sistem, tapi jika data tidak benar maka data akan kembali ke menu awal, setelah proses login berhasil dapat menampilkan halaman utama login, setelah itu proses selesai.



Gambar 4.32 *State Diagram Ticket*

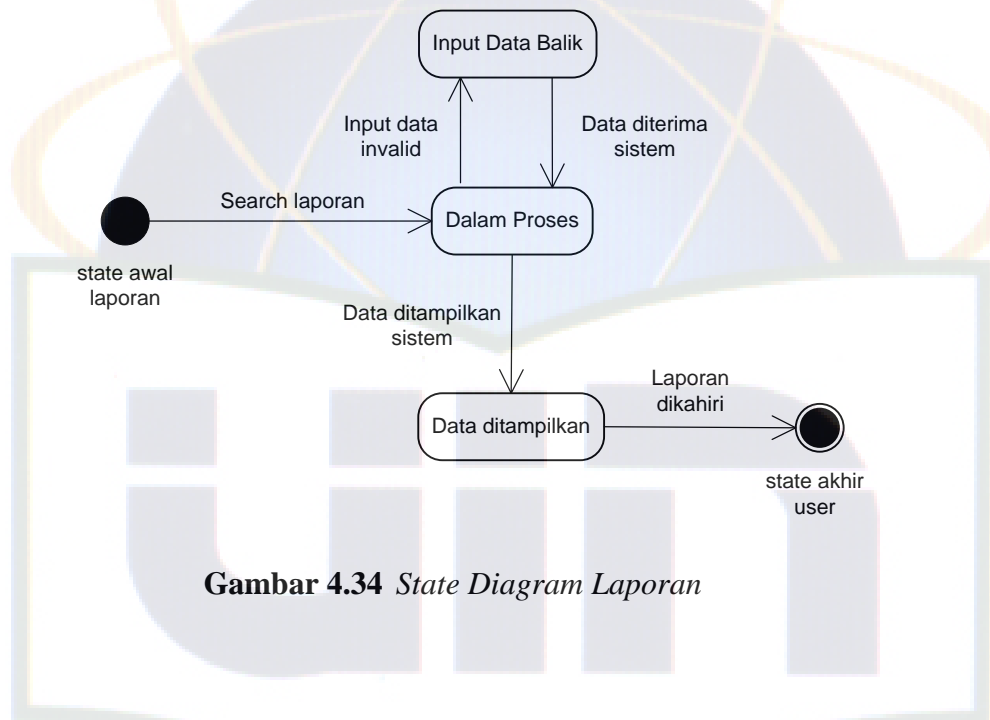
Alur proses nya adalah input ticket di proses jika data benar akan masuk kedalam sistem, tapi jika data salah maka data tidak dapat diproses, kemudian data yang benar di simpan kedalam sistem, kemudian data di assign ticket ke teknisi dan akan menghasilkan solusi, kemudian solusi disimpan kedalam sistem, setelah itu proses selesai.



Gambar 4.33 *State Diagram User*

Alur proses nya adalah user memulai aktifitas kemudian data diproses kedalam sistem, jika data benar, maka akan disimpan kedalam sistem, tapi jika data tidak benar maka tidak dapat diproses, kemudian data yang benar akan

disimpan kedalam database, data dapat ditampilkan, data juga dapat diambil dari database, data juga bisa di ubah, data juga bisa dihapus, setelah itu proses selesai.



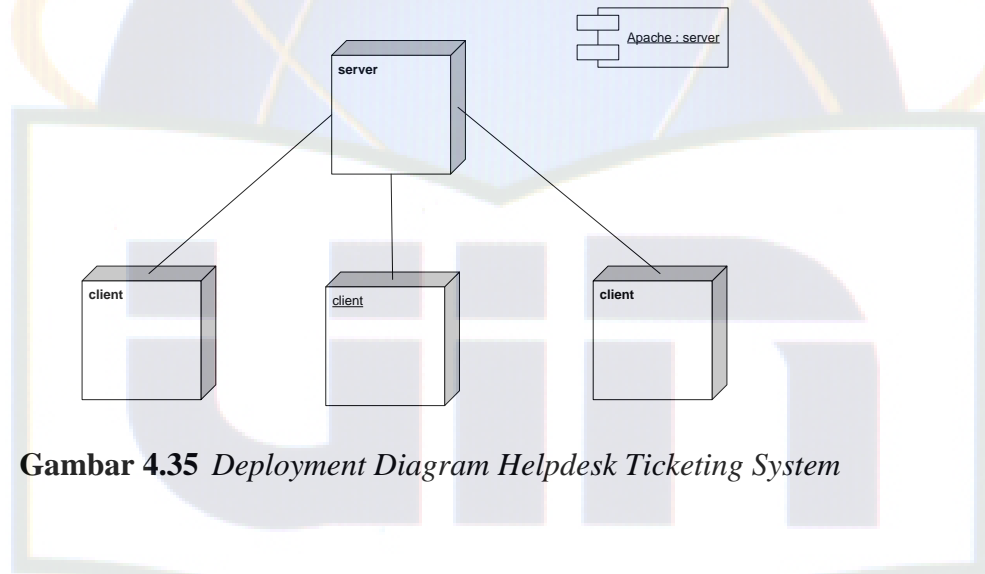
Gambar 4.34 *State Diagram Laporan*

Alur jalanya proses adalah laporan dicari dan dipilih untuk diproses, jika data benar maka data dapat diinput, tapi jika data salah maka data tidak dapat diinput, setelah data yang benar diproses, maka data dapat ditampilkan, setelah itu proses selesai.

4.3.1.5 Pemodelan *Deployment Diagram*

Deployment diagram menunjukkan susunan fisik jaringan dan letak dimana komponen-komponen jaringan.

Pada Gambar 4.14 menggambarkan *deployment diagram* pada *HelpDesk Ticketing System*.



Gambar 4.35 *Deployment Diagram Helpdesk Ticketing System*

Deployment diagram tersebut menggambarkan komponen-komponen yang terlibat dalam pengimplementasian sistem nantinya, yaitu *server* aplikasi yang di dalamnya terdapat *Apache web server*, dengan bahasa pemrograman PHP, dan *database MySQL*, dihubungkan dengan PC *Client* dengan jaringan komunikasi HTTP *connection*.

4.3.2 Desain *Database Sistem*

4.3.2.1 *Physical Data Model (PDM)*

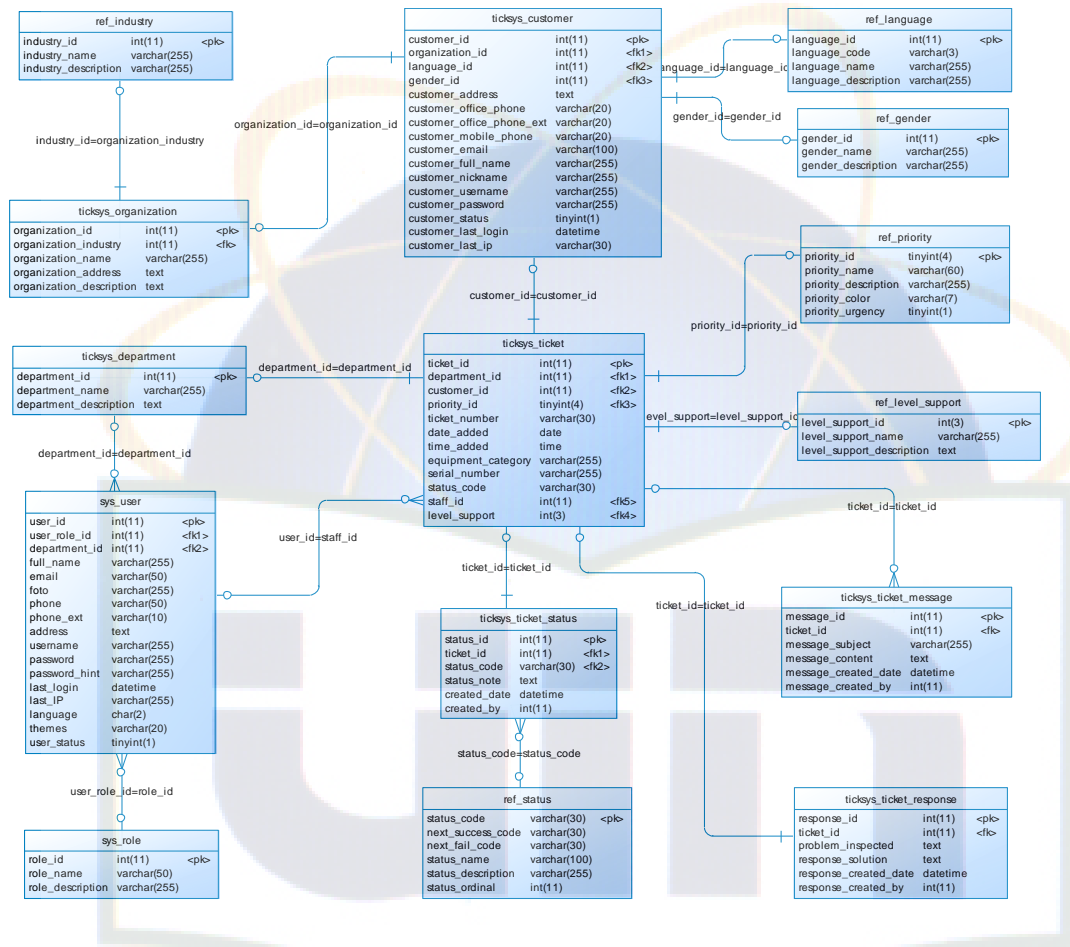
Gambaran PDM nantinya merupakan gambaran *database* yang akan dikonstruksi pada *Helpdesk Ticketing System*. PDM menggambarkan hubungan

antar tabel pada *database* dengan adanya *foreign key* pada suatu tabel yang merupakan *primary key* pada tabel lainnya.

Terdapat 15 tabel yang akan dikonstruksi pada *Helpdesk Ticketing System* ini, yaitu:

1. Tabel *ref_gender*.
2. Tabel *ref_industry*.
3. Tabel *ref_language*.
4. Tabel *ref_level_support*.
5. Tabel *ref_priority*.
6. Tabel *ref_status*.
7. Tabel *sys_role*.
8. Tabel *sys_user*.
9. Tabel *ticksys_Client*.
10. Tabel *ticksys_department*.
11. Tabel *ticksys_organization*.
12. Tabel *ticksys_ticket*.
13. Tabel *ticksys_message*.
14. Tabel *ticksys_response*.
15. Tabel *ticksys_status*.

Gambar 4.36 adalah gambar PDM yang merupakan hasil *generate* dari *class diagram*.



Gambar 4.36 Physical Data Model Helpdesk Ticketing System

Pemodelan PDM pada Gambar 4.15 memperlihatkan hubungan yang terjadi antar tabel-tabel yang terdapat pada *database Helpdesk Ticketing System*. Tabel *sys_user* berhubungan dengan tabel *sys_role* dengan *foreign key* *user_role_id* dan tabel *ticksys_department* dengan *foreign key* *department_id* yang berguna untuk memverifikasi data pengguna dalam proses *login*. Tabel *ticksys_organization* berhubungan dengan tabel *ref_industry* dengan *foreign key* *organization_industry* yang berguna untuk menyimpan data perusahaan. Tabel *ticksys_Client* berhubungan dengan tabel *ticksys_organization* dengan *foreign_key* *organization_id*, tabel *ref_language* dengan *foreign_key* *language_id* dan tabel *ref_gender* dengan *foreign_key* *gender_id* yang berguna untuk menyimpan data

Client. Tabel *ticksys_ticket* berhubungan dengan tabel *ticksys_department* dengan *foreign_key* *department_id*, tabel *ticksys_Client* dengan *foreign_key* *Client_id*, tabel *ref_priority* dengan *foreign_key* *priority_id*, tabel *ref_level_support* dengan *foreign_key* *level_support*, tabel *sys_user* dengan *foreign_key* *staff_id* yang berguna dalam proses pencatatan data *ticket*. Tabel *ticksys_message* berhubungan dengan tabel *ticksys_ticket* dengan *foreign_key* *ticket_id* yang berguna untuk mencatat data *message ticket*. Tabel *ticksys_status* berhubungan dengan tabel *ticksys_ticket* dengan *foreign_key* *ticket_id* dan tabel *ref_status* dengan *foreign_key* *status_code* yang berguna untuk mencatat *history status ticket*. Tabel *ticksys_response* berhubungan dengan tabel *ticksys_ticket* dengan *foreign_key* *ticket_id* yang berguna untuk mencatat *response* atau solusi dari *engineer*.

4.3.2.2 Data Definition Language (DDL)

Dalam pengimplementasian *database* sistem diperlukan spesifikasi tabel-tabel yang terdapat dalam *database*. Spesifikasi *database* ini akan ditentukan dengan DDL yang disimpan dalam suatu kamus data. Kamus data ini berguna bagi DBMS untuk membangun atau membuat tipe data, *field* dan struktur hubungan.

1. Tabel *Gender*

Nama Tabel	: <i>ref_gender</i>
Tipe File	: <i>master file</i>
Primary Key	: <i>gender_id</i>
Foreign Key	: -

Tabel 4.23 Struktur Tabel *Gender*

No	Nama Field	Type	Length	Null	Default
1	<u>gender_id</u>	int	11	No	
2	gender_name	varchar	255	No	
3	gender_description	varchar	255	Yes	

2. Tabel *Industry*

Nama Tabel : ref_industry

Type File : *master file*

Primary Key : industry_id

Foreign Key : -

Tabel 4.24 Struktur Tabel *Industry*

No	Nama Field	Type	Length	Null	Default
1	<u>industry_id</u>	int	11	No	
2	industry_name	varchar	255	No	
3	industry_description	varchar	255	Yes	

3. Tabel *Language*

Nama Tabel : ref_language

Type File : *master file*

Primary Key : language_id

Foreign Key : -

Tabel 4.25 Struktur Tabel *Language*

No	Nama Field	Type	Length	Null	Default
1	<u>language_id</u>	int	11	No	
2	language_code	varchar	3	No	
3	language_name	varchar	255	No	
4	language_description	varchar	255	Yes	

4. Tabel *Level Support*

Nama Tabel : ref_level_support

Tipe File : master file

Primary Key : industry_id

Foreign Key : -

Tabel 4.26 Struktur Tabel *Level Support*

No	Nama Field	Type	Length	Null	Default
1	<u>level_support_id</u>	int	3	No	
2	level_support_name	varchar	255	No	
3	level_support_description	text	0	Yes	

5. Tabel *Priority*

Nama Tabel : ref_priority

Tipe File : master file

Primary Key : priority_id

Foreign Key : -

Tabel 4.27 Struktur Tabel *Priority*

No	Nama Field	Type	Length	Null	Default
1	<u>priority_id</u>	tinyint	4	No	
2	priority_name	varchar	60	No	
3	priority_description	varchar	255	Yes	
4	priority_color	varchar	7	No	
5	priority_urgency	tinyint	1	No	

6. Tabel *Status*

Nama Tabel : ref_status

Tipe File : master file

Primary Key : status_code

Foreign Key : -

Tabel 4.28 Struktur Tabel *Status*

No	Nama Field	Type	Length	Null	Default
1	<u>status_code</u>	varchar	30	No	
2	next_success_code	varchar	30	No	
3	next_fail_code	varchar	30	No	
4	status_name	varchar	100	No	
5	status_description	varchar	255	Yes	
6	status_ordinal	int	5	No	

7. Tabel *Role*

Nama Tabel : sys_role

Tipe File : security file

Primary Key : role_id

Foreign Key : -

Tabel 4.29 Struktur Tabel *Role*

No	Nama Field	Type	Length	Null	Default
1	<u>role_id</u>	int	11	No	
2	role_name	varchar	50	No	
3	role_description	varchar	255	Yes	

8. Tabel *User*

Nama Tabel : sys_user

Tipe File : security file

Primary Key : user_id

Foreign Key : user_role_id, department_id

Tabel 4.30 Struktur Tabel *User*

No	Nama Field	Type	Length	Null	Default
1	<u>user_id</u>	int	11	No	
2	user_role_id	int	11	No	
3	department_id	int	11	No	
4	full_name	varchar	255	No	
5	email	varchar	255	No	
6	foto	varchar	255	No	

7	phone	varchar	255	No	
8	phone_ext	varchar	255	No	
9	address	text	0	No	
10	username	varchar	255	No	
11	password	varchar	255	No	
12	password_hint	varchar	255	No	
13	last_login	datetime	0	No	
14	last_IP	varchar	255	No	
15	language	char	2	No	
16	themes	varchar	255	No	
17	user_status	tinyint	1	No	

9. Tabel *Client*

Nama Tabel : ticksys_Client

Tipe File : work file

Primary Key : user_id

Foreign Key : organization_id, language_id, gender_id

Tabel 4.31 Struktur Tabel *Client*

No	Nama Field	Type	Length	Null	Default
1	<u>Client_id</u>	int	11	No	
2	organization_id	int	11	No	
3	language_id	int	11	No	
4	gender_id	int	11	No	
5	Client_address	text	0	No	
6	Client_office_phone	varchar	20	No	
7	Client_office_phone_ext	varchar	20	No	
8	Client_mobile_phone	varchar	25	No	
9	Client_email	varchar	100	No	
10	Client_full_name	varchar	255	No	
11	Client_nickname	varchar	255	No	
12	Client_username	varchar	255	No	
13	Client_password	datetime	0	No	
14	Client_status	tinyint	1	No	
15	Client_last_login	datetime	0	No	
16	Client_last_ip	varchar	30	No	

10. Tabel *Department*

Nama Tabel : ticksys_department

Tipe File : master file

Primary Key : department_id

Foreign Key : -

Tabel 4.32 Struktur Tabel *Department*

No	Nama Field	Type	Length	Null	Default
1	<u>department_id</u>	int	11	No	
2	department_name	varchar	255	No	
3	department_description	text	0	Yes	

11. Tabel *Organization*

Nama Tabel : ticksys_organization

Type File : master file

Primary Key : organization_id

Foreign Key : organization_industry

Tabel 4.33 Struktur Tabel *Organization*

No	Nama Field	Type	Length	Null	Default
1	<u>organization_id</u>	int	11	No	
2	organization_name	varchar	255	No	
3	organization_industry	int	11	No	
4	organization_address	text	0	No	
5	organization_description	text	0	Yes	

12. Tabel *Ticket*

Nama Tabel : ticksys_ticket

Type File : work file

Primary Key : user_id

Foreign Key : department_id, priority_id, Client_id

Tabel 4.34 Struktur Tabel *Ticket*

No	Nama Field	Type	Length	Null	Default
1	<u>ticket_id</u>	int	11	No	
2	ticket_number	varchar	30	No	
3	date_added	date	0	No	

4	time_added	time	0	No	
5	department_id	int	11	No	
6	priority_id	int	11	No	
7	Client_id	int	11	No	
8	equipment_category	varchar	255	No	
9	serial_number	varchar	255	No	
10	status_code	varchar	30	No	
11	staff_id	int	11	No	
12	level_of_support	int	3	No	

13. Tabel *Message*

Nama Tabel : ticksys_ticket_message

Tipe File : *work file*

Primary Key : message_id

Foreign Key : ticket_id

Tabel 4.35 Struktur Tabel *Message*

No	Nama Field	Type	Length	Null	Default
1	<u>message_id</u>	int	11	No	
2	ticket_id	int	11	No	
3	message_subject	varchar	255	No	
4	message_content	text	0	No	
5	message_created_date	datetime	0	No	
6	message_created_by	int	11	No	

14. Tabel *Response*

Nama Tabel : ticksys_response

Tipe File : *work file*

Primary Key : user_id

Foreign Key : ticket_id

Tabel 4.36 Struktur Tabel *Response*

No	Nama Field	Type	Length	Null	Default
1	<u>response_id</u>	int	11	No	
2	ticket_id	int	11	No	
3	problem_inspected	text	0	No	
4	response_solution	text	0	No	

5	response_created_date	datetime	0	No	
6	response_created_by	int	11	No	

15. Tabel *Ticket Status*

Nama Tabel : ticlsys_ticket_status

Tipe File : *work file*

Primary Key : status_id

Foreign Key : ticket_id, status_code

Tabel 4.37 Struktur Tabel *Ticket Status*

No	Nama Field	Type	Length	Null	Default
1	<u>status_id</u>	int	11	No	
2	ticket_id	int	11	No	
3	status_code	varchar	30	No	
4	satus_note	text	0	No	
5	created_date	datetime	0	No	
6	created_by	int	11	No	

5. Matriks *Data-to-Location-CRUD*

Tabel 4.38 menggambarkan *matriks data-to-location* (CRUD) dari *database Helpdesk Ticketing System*:

Tabel 4.38 Matriks *Data-to-Location-CRUD*

Entry-Attribute	Location	Frontdesk	Client	Engineer	Manager
<i>ref_gender</i>					
gender_id		R	X	X	X
gender_name		CRUD	X	X	X
gender_description		CRUD	X	X	X

<i>ref_industry</i>				
industry_id	R	X	X	X
industry_name	CRUD	X	X	X
industry_description	CRUD	X	X	X
<i>ref_language</i>				
language_id	R	X	X	X
language_code	CRUD	X	X	X
language_name	CRUD	X	X	X
language_description	CRUD	X	X	X
<i>ref_level_support</i>				
level_support_id	R	X	X	X
level_support_name	CRUD	X	X	X
level_support_description	CRUD	X	X	X
<i>ref_priority</i>				
priority_id	R	X	X	X
priority_name	CRUD	X	X	X
priority_description	CRUD	X	X	X
priority_color	CRUD	X	X	X
priority_urgency	CRUD	X	X	X
<i>ref_status</i>				
status_code	R	X	X	X
next_success_code	CRUD	X	X	X
next_fail_code	CRUD	X	X	X
status_name	CRUD	X	X	X
status_description	CRUD	X	X	X
status_ordinal	CRUD	X	X	X
<i>sys_role</i>				
role_id	R	X	X	X
role_name	CRUD	X	X	X
role_description	CRUD	X	X	X
<i>sys_user</i>				

user_id	R	X	R	R
user_role_id	R	X	R	R
department_id	R	X	R	R
full_name	CRUD	X	CRUD	CRUD
email	CRUD	X	CRUD	CRUD
foto	CRUD	X	CRUD	CRUD
phone	CRUD	X	CRUD	CRUD
phone_ext	CRUD	X	CRUD	CRUD
address	CRUD	X	CRUD	CRUD
username	R	X	R	R
password	RUD	X	RUD	RUD
password_hint	CRUD	X	CRUD	CRUD
last_login	X	X	X	X
last_IP	X	X	X	X
language	X	X	X	X
themes	X	X	X	X
user_status	CRU	X	CRU	CRU
<i>ticksys_Client</i>				
Client_id	R	R	R	R
organization_id	R	R	R	R
language_id	R	RU	R	R
gender_id	R	CRUD	R	R
Client_address	R	CRUD	R	R
Client_office_phone	R	CRUD	R	R
Client_office_phone_ext	R	CRUD	R	R
Client_mobile_phone	R	CRUD	R	R
Client_email	R	CRUD	R	R
Client_full_name	R	CRUD	R	R
Client_nickname	R	CRUD	R	R
Client_username	R	R	R	R
Client_password	X	RU	X	X

Client_status	X	R	X	X
Client_last_login	X	R	X	X
Client_last_ip	X	R	X	X
<i>ticksys_department</i>				
department_id	R	X	X	X
department_name	CRUD	X	X	X
department_description	CRUD	X	X	X
<i>ticksys_organization</i>				
organization_id	R	R	R	R
organization_name	CRUD	R	R	R
organization_industry	RU	R	R	R
organization_address	CRUD	R	R	R
organization_description	CRUD	R	R	R
<i>ticksys_ticket</i>				
ticket_id	R	R	R	R
ticket_number	R	R	R	R
date_added	R	R	R	R
time_added	R	R	R	R
department_id	R	RU	R	R
priority_id	RU	R	R	R
Client_id	R	R	R	R
equipment_category	R	CRUD	R	R
serial_number	R	CRUD	R	R
status_code	R	R	R	R
staff_id	RU	R	R	R
level_of_support	CRU	R	R	R
<i>ticksys_ticket_message</i>				
message_id	R	R	R	R
ticket_id	R	R	R	R
message_subject	R	CRUD	R	R
message_content	R	CRUD	R	R

message_created_date	R	R	R	R
message_created_by	R	R	R	R
<i>ticksys_ticket_response</i>				
response_id	R	R	R	R
ticket_id	R	R	R	R
problem_inspected	R	R	CRUD	R
response_solution	R	R	CRUD	R
response_created_date	R	R	R	R
response_created_by	R	R	R	R
<i>ticksys_ticket_status</i>				
status_id	R	R	R	R
ticket_id	R	R	R	R
status_code	R	R	R	R
satus_note	CR	R	R	R
created_date	R	R	R	R
created_by	R	R	R	R

Keterangan:

C: *Create* R: *Read* U: *Update*

D: *Delete* X: *No access*

4.3.3 Desain Antarmuka Sistem

Dalam mengembangkan perangkat lunak *Helpdesk Ticketing System* diperlukan gambaran desain antarmuka sebagai petunjuk pembuatan antarmuka perangkat lunak.

Gambaran antarmuka *Helpdesk Ticketing System* tersebut dapat dilihat pada Lampiran 3.

4.4 System Implementation (Implementasi Sistem)

4.4.1 Konstruksi Perangkat Lunak

Blueprint (cetak biru) sistem yang telah dimodelkan dan dirancang sebelumnya dieksekusi menjadi sebuah set kode program dengan menggunakan bahasa pemrograman PHP guna mengimplementasikan *Helpdesk Ticketing System*.

Sejumlah *tools* digunakan untuk mengembangkan perangkat lunak *Helpdesk Ticketing System*. Pada tabel berikut ditunjukkan daftar *tools* yang digunakan.

Tabel 4.39 Daftar *Tools* Pengembangan Perangkat Lunak Sistem

No.	<i>Tool</i>	Kegunaan
1	<i>Power Designer 12</i>	Menggambar pemodelan berorientasi obyek.
2	<i>Microsoft Office Visio 2007</i>	Menggambar pemodelan berorientasi obyek.
3	PHP 5.2.9	Bahasa pengembangan untuk mengembangkan Sistem Informasi Produksi Percetakan.
4	<i>MySQL 5.0.51</i>	<i>Database</i> yang digunakan dalam sistem.
5	<i>Eclipse Galileo</i>	Perangkat <i>editor</i> pembuatan <i>program</i> .
6	XAMPP <i>Apache HTTP Server</i>	<i>Web server</i> .
7	<i>PhpMyAdmin</i>	Perangkat untuk mengakses <i>database</i> yang terdapat pada XAMPP <i>Apache</i> .
8	<i>Navicat Premium 8.2</i>	Perangkat untuk mengakses <i>database</i> .
9	<i>Mozilla Firefox 3.5</i>	<i>Web browser</i> .

Terdapat beberapa modul yang utama untuk dikonstruksi pada perangkat lunak *Helpdesk Ticketing System*, yaitu:

1. Modul data *master* yang terdiri dari *master gender*, *master industry*, *master language*, *master level_support*, *master priority* dan *master status*.
2. Modul *organization*.
3. Modul *Client*.
4. Modul *ticket*.
5. Modul *profile*.
6. Modul ubah *password*.

4.4.2 Pengujian Perangkat Lunak

4.4.2.1 Alpha Testing

Setelah perangkat lunak *Helpdesk Ticketing System* dikonstruksi, tahap selanjutnya adalah melakukan pengujian terhadap perangkat lunak sistem. Pada pengujian sistem, *testing* yang dilakukan menggunakan metode *blackbox testing*. Cara pengujian hanya dilakukan dengan menjalankan atau mengeksekusi unit atau modul, kemudian diamati apakah hasil dari unit itu sesuai dengan proses bisnis yang diinginkan. Pada *blackbox testing* ini, yang dilakukan hanyalah *input* data dan dilihat apakah *output*-nya sesuai dengan proses bisnis yang diharapkan.

Pengujian *black box testing* yang pertama dilakukan dengan menggunakan data *dummy* atau percobaan untuk memastikan bahwa tidak terjadi kesalahan atau *error* pada pemrograman perangkat lunak sistem. Pengujian *black box testing* ini adalah merupakan *alpha testing*.

Alpha testing dilakukan sendiri oleh penulis. Setelah dilakukan pengujian ini, penulis menyatakan bahwa tidak terdapat *error* pada pemrograman perangkat lunak *Helpdesk Ticketing System*.

4.4.2.2 Beta Testing

Setelah memastikan bahwa tidak terdapat *error* dalam pemrograman perangkat lunak *Helpdesk Ticketing System* dengan melakukan *alpha testing*. Tahap selanjutnya adalah melakukan *beta testing*.

Beta testing dilakukan dengan menggunakan metode yang sama dengan *alpha testing* yaitu *black box testing*. Hanya saja data yang dipergunakan dalam pengujian ini adalah data sebenarnya yang terdapat pada *Helpdesk Ticketing System*. Pengujian ini dimaksudkan untuk mengetahui apakah *output* yang dihasilkan dari pengolahan sistem benar-benar sesuai dengan *output* yang diharapkan oleh pengguna *Helpdesk Ticketing System*.

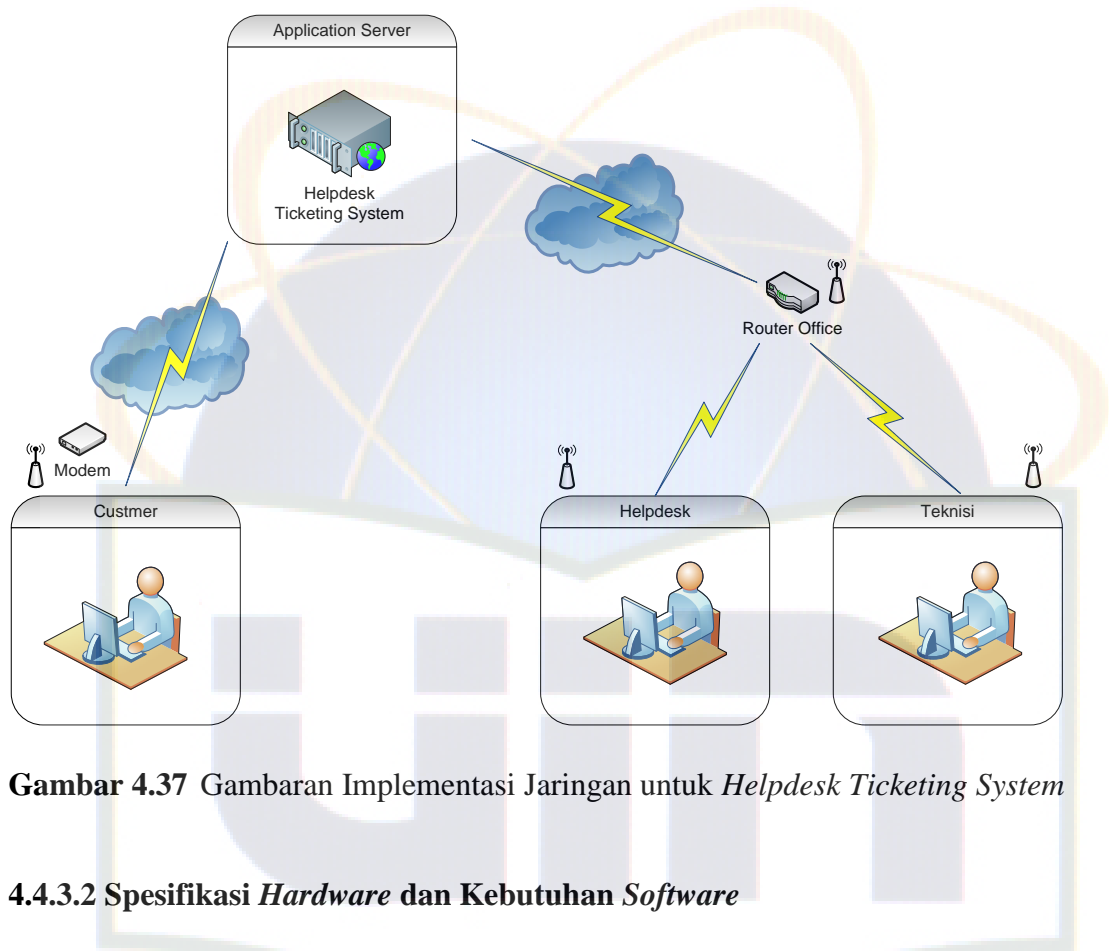
Setelah melakukan *beta testing* dan melihat hasil pengujian dapat disimpulkan bahwa perangkat lunak *Helpdesk Ticketing System* telah sesuai dengan proses bisnis dalam *Helpdesk Ticketing System* dan telah menghasilkan *output* yang sesuai dengan kebutuhan.

Hasil *beta testing* dapat dilihat pada Lampiran 4.

4.4.3 Implementasi Perangkat Lunak

4.4.3.1 Penyiapan Rencana Implementasi Jaringan

Helpdesk Ticketing System yang telah dibuat akan diimplementasikan melalui jaringan Internet. Gambaran implementasi jaringan yang diterapkan menggunakan jaringan internet dapat dilihat pada Gambar 4.37.



Gambar 4.37 Gambaran Implementasi Jaringan untuk *Helpdesk Ticketing System*

4.4.3.2 Spesifikasi *Hardware* dan Kebutuhan *Software*

Dalam pengimplementasian *Helpdesk Ticketing System*, penulis mengusulkan penggunaan *hardware* dan *software* dengan spesifikasi dan kebutuhan sebagai berikut:

1. *Hardware*

Pada Tabel 4.40 berikut ini dapat dilihat seperti apa spesifikasi *hardware* untuk menjalankan perangkat lunak *Helpdesk Ticketing System* secara *online*.

Tabel 4.40 Spesifikasi *Hardware* untuk Implementasi Perangkat Lunak Sistem

<i>SERVER</i>		
<i>ENVIRONMENT</i>	<i>EQUIPMENT</i>	<i>DESCRIPTION</i>
PC	Desktop / Workstation	<ul style="list-style-type: none"> Processor Intel Core 2 Duo 2,8GHz Memory 4GB DR2 PC 5300 (2x

		2GB) <ul style="list-style-type: none">• Hard-disk 320GB SATA 7200rpm
Network	Internet	
Input Devices	Keyboard, Mouse	
Output Devices	Monitor LCD Flat 17 inch, Printer	
CLIENT		
ENVIRONMENT	EQUIPMENT	DESCRIPTION
PC/Laptop	Desktop / Workstation/Laptop	<ul style="list-style-type: none">• Processor Intel Pentium 4 2,0 GHz• Memory 2GB DR2 PC 5300• Wifi + Network LAN
Network	Internet	
Input Devices	Keyboard, Mouse	
Output Devices	Monitor, Printer	

2. *Software*

Berikut ini merupakan *software* yang dibutuhkan dalam menjalankan *Helpdesk Ticketing System*:

- Operating System: Windows XP SP2 or above.*
- Apache HTTP Server 2.2.9.*
- MySQL Server 5.0 or above.*
- Mozilla Firefox 3.0 (or above), Opera 10.0 (or above), Chrome, IE 7 (or above), Safari 4 (or above).*

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pembahasan yang dilakukan pada *HelpDesk Ticketing System*, maka di peroleh beberapa kesimpulan:

- a. Terkontrolnya proses dalam memperbaiki keluhan *Client* internet PT. Primus Indojoya dan penanganan kerusakan *hardware* terkoordinasi dengan baik.
- b. Pencatatan masalah dapat dilakukan secara sistematis, sehingga data aman dan akurat dengan tidak adanya data yang duplikasi. Dengan adanya *HelpDesk Ticketing System*, maka tugas Kepala Bidang IT lebih terkomputerisasi dalam melihat laporan bulanan permasalahan yang dialami *Client*, baik masalah koneksi jaringan internet maupun kerusakan *hardware*.
- c. Frekuensi masalah dapat di lihat langsung melalui sistem ini, tidak harus di hitung manual. Hal ini dapat memudahkan pekerjaan Kepala Bidang IT dalam membuat keputusan. Adanya grafik tiap bulan yang menggambarkan fluktuasi permasalahan baik terkait koneksi jaringan internet dan kerusakan *hardware* sehingga dapat melihat sejauh mana tingkat masalah yang ditangani serta pemanfaatan kinerja *helpdesk* dalam menangani permasalahan yang ada.

5.2 Saran

Saran-saran yang sebaiknya dilakukan *HelpDesk Ticketing System*, yaitu sebagai berikut:

- a. Lingkup penanganan masalah dapat diperluas, sehingga Kepala Bidang IT mendapatkan gambaran secara luas tentang permasalahan yang terjadi di perusahaan tersebut.
- b. Laporan dapat disajikan lebih detail, yakni dengan pemantauan masalah laporan per-hari atau per-minggu supaya permasalahan yang terjadi lebih termonitor.
- c. Dalam mengukur tingkat masalah, analisa tidak hanya dilihat dari jenis masalah saja tetapi sesuai dengan ukuran yang menjadi harapan bagi penunjang keputusan hasil.
- d. Untuk mengantisipasi bila terjadi kerusakan *software*, maka secara berkala mengontrol data-data laporan permasalahan dengan menitikberatkan pada keamanan data.
- e. Perlu melakukan pelatihan untuk meningkatkan kesiapan para pegawai yang akan menggunakan sistem ini, dengan adanya pelatihan akan membantu pegawai agar dapat dengan mudah menggunakan sistem yang baru.

Untuk menghasilkan informasi yang akurat, maka diperlukan ketelitian dalam menginput data. Pegawai memerlukan informasi yang akurat dan ketelitian agar tidak terjadi kesalahan dalam memasukkan data.

DAFTAR PUSTAKA

- Bogs, Wendy, Michael Boggs. *Mastering UML with Rational Rose* 2002. Alamenda: Sybex. 2002.
- Buttle F. 2007. *CRM (Manajemen Hubungan Pelanggan) : Concept and Tools*. Malang: Bayumedia Publishing.
- Fowler, Martin. 2004. *UML Distilled Edisi ke-3*. Yogyakarta: Penerbit Andi.
- Griffin J. 2003. *Customer Loyalty Menumbuhkan dan Mempertahankan Kesetiaan Pelanggan*. Jakarta: Erlangga.
- Gunarsa YSD. 2002. *Asas-asas Psikologi Keluarga Idaman*. Jakarta: Gunung Mulia.
- Hariyanto, Bambang. 2004. *Rekayasa Sistem Berorientasi Objek*. Bandung: Informatika.
- Hariyanto, Bambang. 2004. *Sistem Manajemen Basis Data : Pemodelan, Perancangan, dan Terapannya*. Bandung: Informatika.
- Hartono, Jogiyanto. *Analisis & Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta: Andi, 2005.
- Hendrik. 2002. *Manajemen Kewirausahaan*. Bandung: Informatika.
- Jogiyanto, H.M. 2005. *Analisis & Desain Sistem Informasi : Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta: Penerbit Andi.
- Junaedi IWR, Hartomo KD. 2009. *Analisis Desain CRM (Customer Relationship Management) pada SIM Hotel dengan Metode CBD (Componen-Based Development)*. *Konferensi Nasional Sistem dan Informatika* 202-207 KNS&I09-037.
- Kadir, Abdul. 2003. *Pengenalan Sistem Informasi*. Yogyakarta: Penerbit Andi.

- Kalakota R, Robinson M. 2001. *E-Business 2.0 : Roadmap for Success*. USA: Addison Wesley, Longman Inc.
- Ladjamudin, Al-Bahra bin. 2005. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu.
- Maharsi, Sri. *Pengaruh Perkembangan Teknologi Informasi Terhadap Bidang Akuntansi Manajemen*. Jurnal Akuntansi & Keuangan Vol. 2, No. 2, Nopember 2000.
- McLeod, Raymond. *Sistem Informasi Manajemen. Jilid 2. Ed. Ke-7*. Alih Bahasa. Jakarta: PT. Prenhallindo. 2004.
- Munawar. *Pemodelan Visual dengan UML*. Yogyakarta: Graha Ilmu. 2005.
- Nadlifatin R, Mudjahidin, Mahendrawathi Er. 2009. *Pengukuran Performansi Praktik Pengelolaan Hubungan Pelanggan Dengan Menggunakan Kerangka CRM Scorecard (Studi Kasus: PT. Bank Mandiri (Persero) Tbk*. Institut Teknologi Sepuluh November.
- Nugroho, Adi. *Konsep Pengembangan Sistem Basis Data*. Bandung: Informatika. 2004.
- Rahmayanty N. 2010. *Manajemen Pelayanan Prima*. Yogyakarta: Graha Ilmu.
- Robbins SP, Coulter M. 2004. *Manajemen Edisi Ketujuh Jilid I*. Jakarta: Indeks.
- Saraswati M, Widaningsih I. 2008. *Be Smart Ilmu Pengetahuan Sosial (Geografi, Sejarah, Sosiologi, Ekonomi) untuk Kelas VIII Sekolah Menengah Pertama*. Bandung: Grafindo Media Pratama.
- Sembiring H, Jhony. 2001. *Jaringan Komputer Berbasis Linux*. Jakarta : Elex Media Komputindo.

Turban, Efraim. *Information Technology for Management: Transforming Organizations in the Digital Economy 4th Edition*. United States: John Wiley & Sons, Inc. 2004.

Wahana. 2006. *Menguasai Pemrograman Web dengan PHP5*. Semarang : Andi Offset Yogyakarta dan Wahana Komputer Semarang.

Wahyono, Teguh. *Sistem Informasi (Konsep Dasar, Analisis Desain, dan Implementasi)*. Yogyakarta: Graha Ilmu. 2004.

Whitten, Jeffery L., Lonnie D. Bentley, Kevin C. Dittman. *Metode Desain dan Analisis Sistem. Edisi 6*. Alih bahasa. Yogyakarta: Andi. 2004.

Zikmund WG, McLeod Jr R, Gilbert FW. 2003. *Customer Relationship Management: Integrating Marketing Strategy and Information Technology*. USA: John Wiley and Sons Inc.

<http://www.help-desk-world.com/help-desk.htm>)

<http://www.remedy.com>)

<http://searchCRM.techtarget.com/definition/trouble-ticket>)

<http://kamii-yogyakarta.tripod.com/tools.htm>



LAMPIRAN 1

OBSERVASI

Laporan Hasil Observasi Lapangan

Lokasi : PT. Primus Indojaya

Waktu Observasi : 4 Juni 2011 – 30 Juni 2011

1. Kegiatan PT. Primus Indojaya
 - a. Mulai masuk kerja pada pukul 08.00 WIB.
 - b. Pegawai staf yang datang harus melakukan absensi terlebih dahulu pada daftar absensi yang sudah tersedia, dan semua mulai bekerja sesuai *jobdesk* nya masing-masing.
 - c. Jam pulang kerja pada pukul 17.00 WIB, pegawai staf mengisi kembali jam pulang nya pada daftar absensi staf.
 - d. Permasalahan pada *software* dan *hardware* ditangani langsung oleh kepala bagian teknis.
 - e. Memberikan solusi mengenai *helpdesk*.

Jakarta, Juni 2011

Kepala Bidang IT

Alex Kurniawan, S.Kom



LAMPIRAN 2

WAWANCARA

WAWANCARA

Wawancara I

Narasumber : Alex Kurniawan
Jabatan : Kepala IT Helpdesk
Penanya : Qoyyimah
Waktu : 04 April 2011
Tema : Prosedur sistem berjalan.
Tujuan : Mengetahui alur kerja dan sistem *helpdesk* yang sedang berjalan.

Pertanyaan:

1. Bagaimana alur kerja sistem *helpdesk* yang sedang berjalan pada PT. Primus Indojaya?
2. Bagaimana memenuhi kebutuhan pelanggan menggunakan aplikasi *ticketing system*?
3. Apakah pengelolaan sistem *helpdesk* yang sedang berjalan sudah terkomputerisasi?
4. Bagaimana membuat aplikasi *ticketing system* yang *userfriendly* dan menarik sehingga mempermudah pelanggan untuk mengaksesnya ?

Hasil Wawancara:

1. Alur sistem *helpdesk* saat ini:
 - Sistem yang berjalan sudah cukup baik, sudah mempunyai banyak pelanggan juga, sehingga sistem yang ada harus lebih dikembangkan lagi untuk menarik pelanggan lebih.
2. Kebutuhan pelanggan menggunakan *ticketing system*:
 - Sangat berpengaruh, karena dengan adanya *ticketing system* proses yang berjalan lebih cepat dan modern.
3. belum, sistem *helpdesk* yang berjalan masih manual, oleh karena itu sistem *helpdesk* disini akan dikembangkan lagi untuk menuju sistem yang terkomputerisasi.
4. Dengan membuat sistem yang menarik dan canggih sehingga mempermudah pelanggan untuk mengaksesnya.



LAMPIRAN 3

SURAT PENELITIAN

SURAT KETERANGAN PENELITIAN

Assalamu'alaikum Wr. Wb

Yang bertanda tangan dibawah ini,

Nama : Alex Kurniawan, S.Kom

Jabatan : Kepala IT Helpdesk PT. Primus Indojoya

Menerangkan bahwa,

Nama : Qoyyimah

NIM : 204093002662

Mahasiswa Program Studi Sistem Informasi

Universitas Islam Negeri Syarif Hidayatullah Jakarta

Telah mengadakan penelitian, observasi, dan wawancara dalam rangka untuk kepentingan skripsi yang berjudul ***Rancang Bangun Helpdesk Ticketing System Dengan Pendekatan Customer Relationship Management (CRM) Studi Kasus: PT. Primus Indojoya.***

Demikian surat keterangan ini dibuat dengan sebenar-benarnya dan dapat digunakan sebagaimana mestinya.

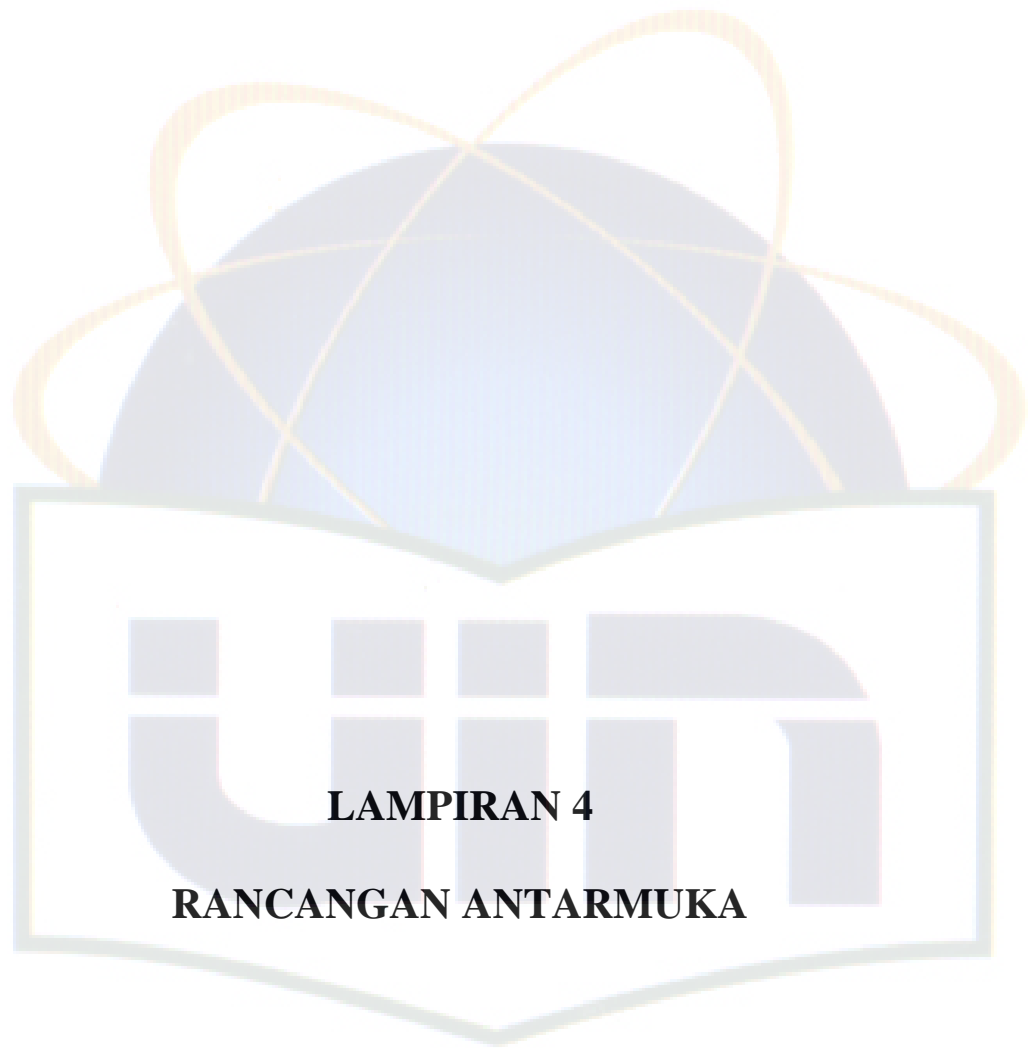
Wassalamu'alaikum Wr. Wb

Jakarta, 6 Juni 2011

Mengetahui

Kepala Bidang IT

(Alex Kurniawan, S.Kom)



LAMPIRAN 4

RANCANGAN ANTARMUKA

LOGIN

HEADER

LOGIN

Username

Password

Login

Register

FOOTER

Register

HEADER

Register

Organization *

:

Customer Name *

:

Email *

:

Address *

:

Office Phone *

:

Ext

Mobile Phone *

:

eHwyDH

:

Verification *

:

Save

Back

FOOTER

View Ticket

HEADER

Register Success

Thangkyou for registering on Ticketing System.
An Email has been sent to you. Please Check your email for verification.
[Already have your login account?](#)

FOOTER

HOME

User Login [Logout]
HEADER

DROPDOWN MENU

TEKS HOME

Login sebagai: xxx
Login Terakhir: 17 Januari 2011
IP: 127.0.0.1

FOOTER

Tickets

User Login [Logout]

HEADER

DROPDOWN MENU

Tickets

Ticket Number: Or Customer :

Showing: [All Tickets](#) [Open](#) [In Progress](#) [Completed](#) [New Ticket](#)

No	Ticket	Date	Time	Subject	Priority	Status

FOOTER

View Ticket

User Login [Logout]

HEADER

DROPDOWN MENU

View Ticket

Ticket Number : #1003-TKT/05/2011

Status : xxxxx Organization Name : xxxxx

Priority : xxxxx Customer Name : xxxxx

Department : xxxxx Email : xxxxx

Created Date : xxxxx Office Phone : xxxxx Ext: -

Mobile Phone : xxxxx

Subject : xxxxx

Equipment Category	Status	Note	Created Date	Created by
Serial Number : xxxxx	OPEN	xxx	xxx	xxx
Assigned Staff : xxxxx	INPROGRESS	xxx	xxx	xxx
Level of Support : xxxxx	COMPLETED	xxx	xxx	xxx

Thursday, 26 May 2011 22:46 , By Customer

Ticket Content Message

Assigned Ticket

User Login [Logout]

HEADER

DROPDOWN MENU

View Ticket

Ticket Number : #1003-TKT/05/2011

Edit Ticket

Status : xxxxxx

Organization Name : xxxxx

Priority : xxxxxx

Customer Name : xxxxx

Department : xxxxxx

Email : xxxxx

Created Date : xxxxxx

Office Phone : xxxxx

Ext: -

Mobile Phone : xxxxx

Subject : xxxxxx

Equipment Category : xxxxxx

Serial Number : xxxxxx

Assigned Staff : xxxxxx

Level of Support : xxxxxx

Status	Note	Created Date	Created by
OPEN	xxx	xxx	xxx

Thursday, 26 May 2011 22:46 , By Customer

Ticket Content Message

Assigned to Staff

Response Ticket

User Login [Logout]

HEADER

DROPDOWN MENU

View Ticket

Ticket Number : #1003-TKT/05/2011

Status : xxxxxx

Organization Name : xxxxx

Priority : xxxxxx

Customer Name : xxxxx

Department : xxxxxx

Email : xxxxx

Created Date : xxxxxx

Office Phone : xxxxx

Ext: -

Mobile Phone : xxxxx

Subject : xxxxxx

Equipment Category : xxxxxx

Serial Number : xxxxxx

Assigned Staff : xxxxxx

Level of Support : xxxxxx

Status	Note	Created Date	Created by
OPEN	xxx	xxx	xxx
INPROGRESS	Xxx	Xxx	xxx

Thursday, 26 May 2011 22:46 , By Customer

Ticket Content Message

Post Response

Create Ticket

User Login [Logout]

HEADER

DROPDOWN MENU

New Ticket

Organization *

:

Customer Name *

:

Email *

:

Address *

:

Office Phone *

:

Ext

Mobile Phone *

:

eHwyDH

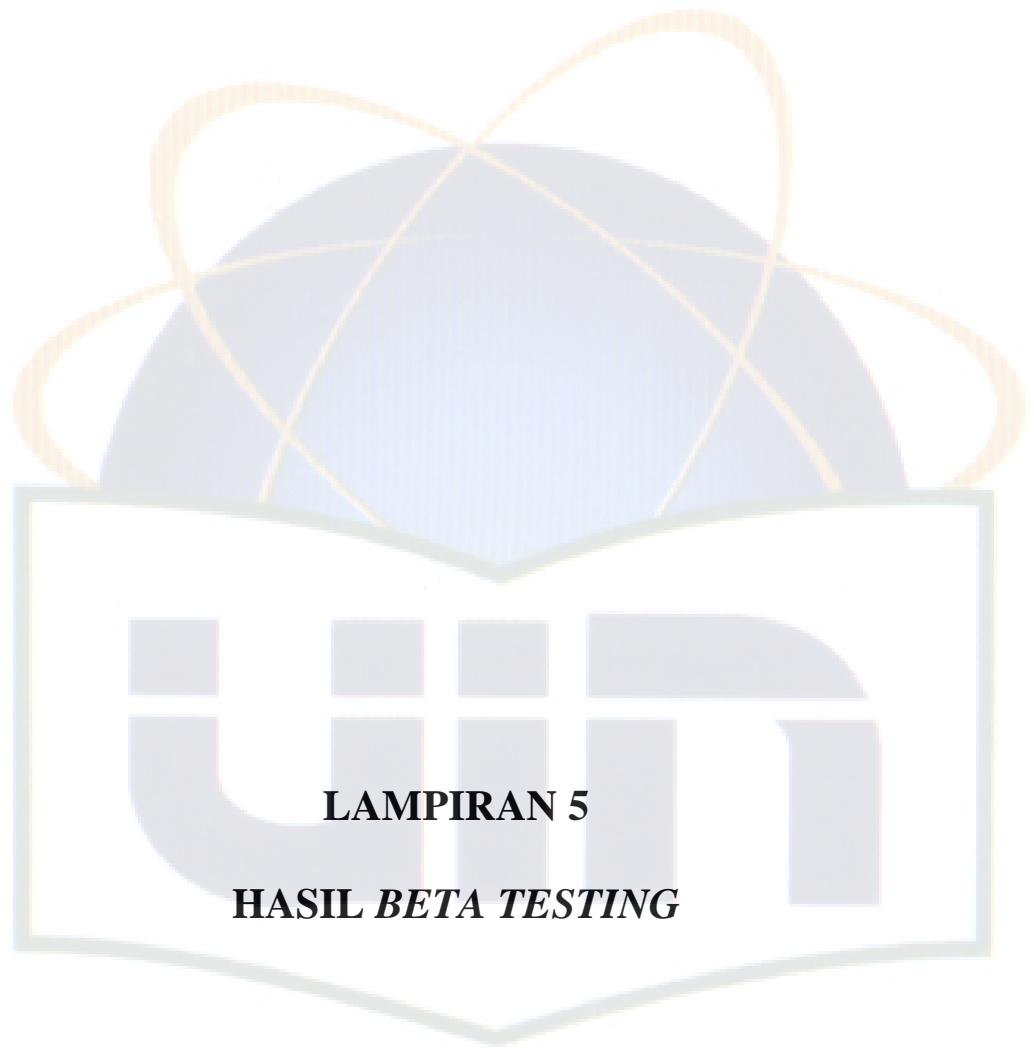
Verification *

:

Save

Back

FOOTER



LAMPIRAN 5

HASIL BETA TESTING

Hasil Beta Testing

Berikut ini adalah hasil *beta testing* terhadap perangkat lunak *Helpdesk Ticketing System*.

Test Case ID	Test Case Name	Description	Expected Result	Test Result
HDTs-BT001	Register	Register: a. User meng-klik menu "Register". b. User memilih nama organisasi, input client name, input email, input address, input office phone, input mobile phone, input verifikasi. c. User meng-klik tombol "Save".	Sistem akan melakukan pengecekan data registrasi terhadap data jika seluruh data sudah benar, data disimpan kedalam sistem.	OK
HDTs-BT002	Login	Login: d. User meng-klik menu "Login". e. User melengkapi form Login dengan memasukkan data disetiap field. f. User meng-klik tombol "Login".	Sistem akan melakukan pengecekan terhadap data yang telah dimasukkan ke dalam form kemudian jika data tersebut benar maka User dapat masuk ke dalam sistem.	OK
HDTs-BT003	Input Ticket	Input Ticket: a. User meng-klik menu "Ticket". b. User memasukkan "detail ticket". c. User memilih menu "Lihat laporan". d. User berhasil memuat "ticket" baru kedalam sistem. e. User meng-klik tombol "Save".	Sistem akan mengecek setiap data yang diinput untuk dicek kebenaran data tersebut, jika data telah terisi dengan benar, simpan data ticket ke dalam sistem dan sistem akan mengirimkan email yang berisi informasi ticket kepada client dan manager.	OK
HDTs-BT004	View Ticket	View Ticket: a. User mengakses menu	Sistem akan menampilkan "List	OK

<i>Test Case ID</i>	<i>Test Case Name</i>	<i>Description</i>	<i>Expected Result</i>	<i>Test Result</i>
		<p>"View Ticket".</p> <p>b. User selesai saat List Ticket telah ditampilkan dan dapat dilihat actor.</p>	Ticket".	
HDTs-BT005	Assign Ticket	<p>Assign Ticket:</p> <p>a. User mengakses "Assign Ticket" dan memilih teknisi yang akan ditugaskan.</p> <p>b. User berakhir saat actor telah berhasil melakukan Assign Ticket.</p>	Sistem akan melakukan pengecekan terhadap data yang akan dimasukkan, jika data yang dimasukkan benar simpan ke dalam sistem, sistem mengirim email peringatan kepada client, teknisi, manager.	OK
HDTs-BT006	Input Solusi	<p>Input Solusi:</p> <p>a. User mengakses menu input solusi dan memasukkan data solusi ticket.</p> <p>b. User telah berhasil melakukan input solusi.</p>	Sistem akan melakukan pengecekan terhadap data yang akan dimasukkan, jika data yang dimasukkan benar simpan ke dalam sistem, sistem mengirim email peringatan kepada client, teknisi, manager.	OK
HDTs-BT007	Lihat Laporan	<p>Lihat Laporan:</p> <p>a. User meng-klik menu "Ticket".</p> <p>b. User meng-klik submenu "Buat Laporan Ticket".</p>	Sistem akan menampilkan halaman ticket, menampilkan halaman laporan ticket, jika laporan data sudah dicek kebenarannya, kemudian laporan diberikan kepada manager.	OK