BAB III

PEMBAHASAN

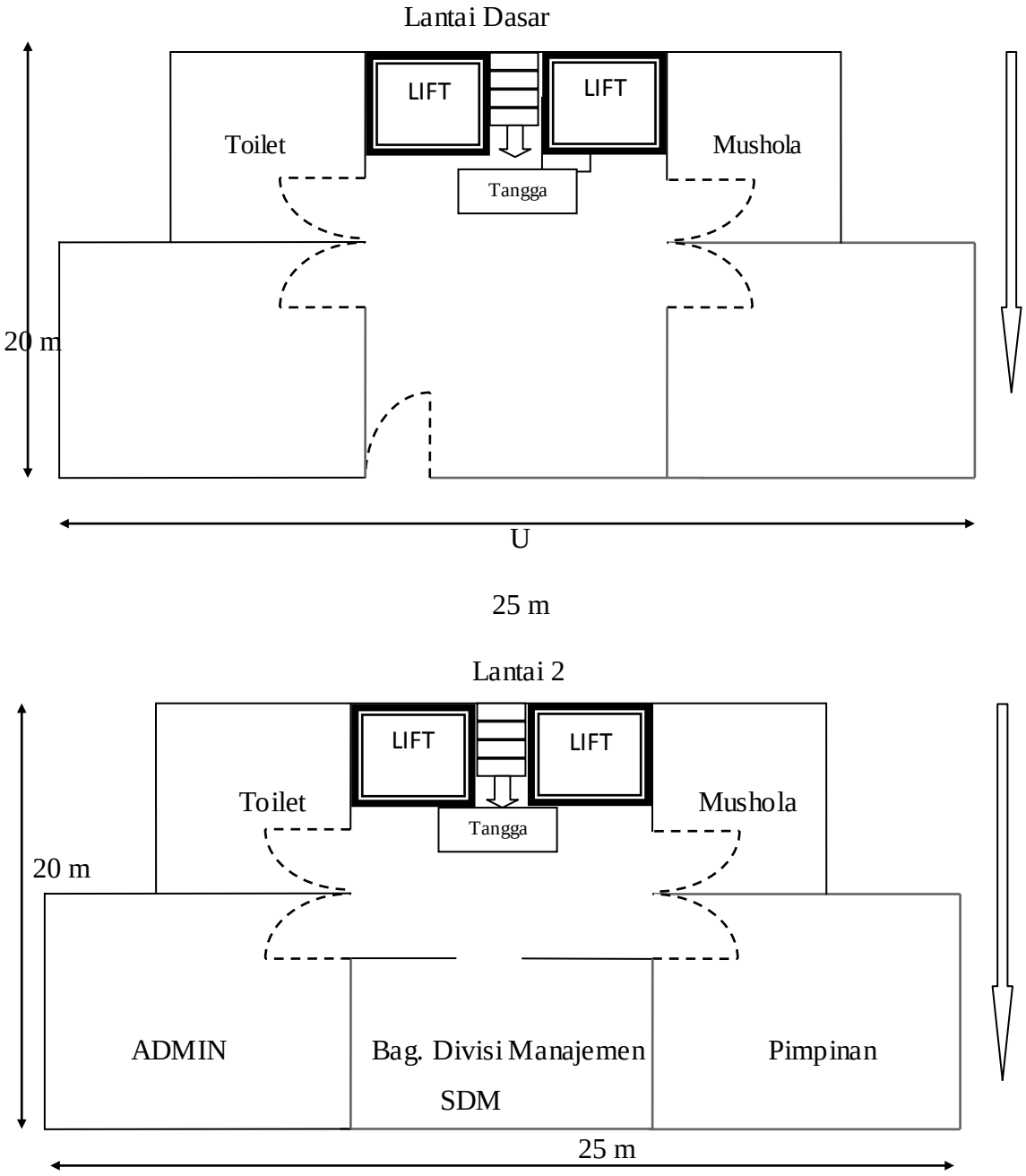
3.1 Analisis Masalah

Dengan semakin bertambahnya kebutuhan *customer* maka bertambah pula jumlah pesanan barang yang di inginkan *customer*,pesanan barang tersebut tidak hanya di wilayah sekitar bandung, tetapi juga ada yang di luar bandung bahkan di luar pulau jawa. Perusahaan harus memberikan pelayanan extra terhadap para pelanggannya, maka perusahaan membangun suatu jaringan *Call Center*. *Customer service* bertugas untuk melayani para *customer* yang memesan barang terhadap perusahaan tersebut yaitu PT. INTI. Sehingga perusahaan dapat mendata pemesan barang secara terperinci, bukan hanya itu dengan adanya *Call Center* ini, perusahaan dapat meningkatkan kinerja dan menghemat waktu.

Jaringan *Call Center* yang dibangun di PT.INTI Divisi Manajemen SDM, sangat membantu para *custumer* yang ingin menyampaikan keluhannya, dengan *call center* yang ada di PT.INTI DIvisi Manajemen SDM ini, *customer* dapat menyampaikan keluhannya lewat jaringan telepon atau sejenisnya dan bias lebih menghemat waktu.

3.2 Analisis Jaringan

Sebelum membangun jaringan di PT.INTI Divisi Manajemen SDM, maka harus diketahui terlebih dahulu denah gedungnya. Gambar 3.1 berikut adalah denah gedung dari PT.INTI Divisi Manajemen SDM :



Gambar 3.1 Denah gedung PT.INTI Divisi Manajemen SDM

Seperti yang tellihat dari denah gedung PT. INTI Divisi Manajemen SDM, maka untuk itu perlu adanya analisa terhadap masalah yang ada dan bagaimana pemecahan masalah dalam membangun jaringan lokal pada PT. INTI Divisi Manajemen SDM

3.2.1 Permasalahan yang dihadapi

Setelah mengetahui denah lokasi dari CV. Asep Hendro Racing Sport (AHRS), terdapat permasalahan yang di hadapi dalam membangun jaringan lokal,yaitu :

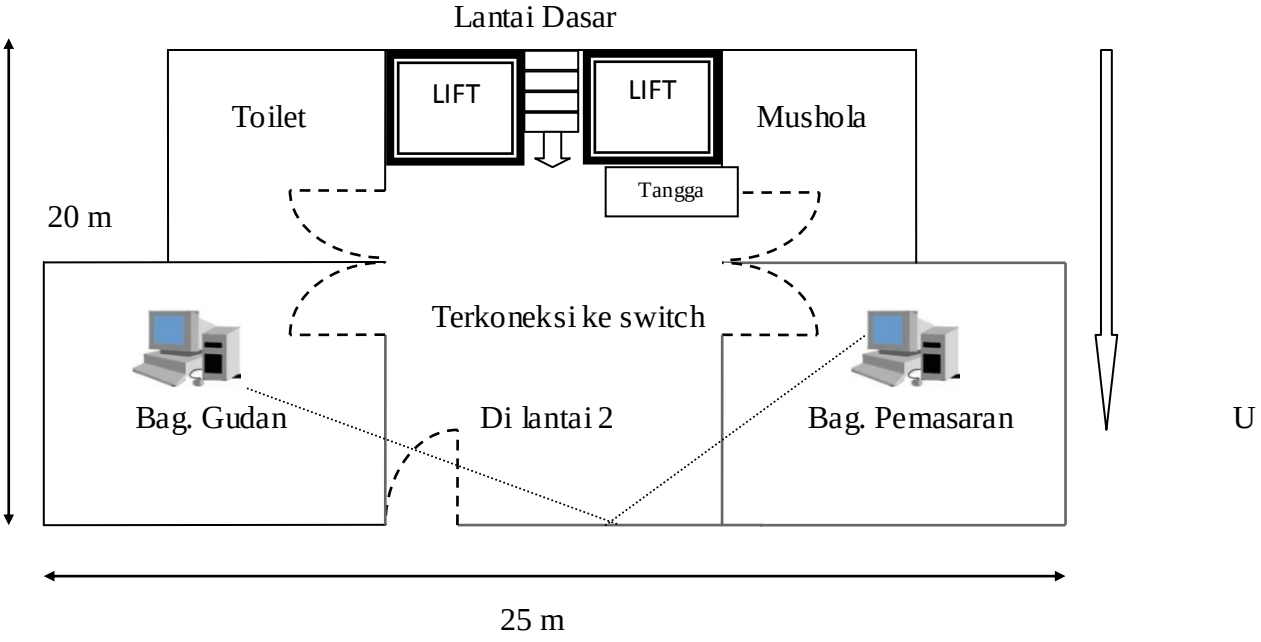
a) Topologi apa yang dipakai untuk menyesuaikan dengan denah lokasi di PT. INTI Divisi Manajemen SDM
b) Apa saja kebutuhan alat yang diperlukan untuk membangun jaringan lokal sesuai denah lokasi di PT. INTI Divisi Manajemen SDM
c) Apa saja kebutuhan alat yang diperlukan untuk membangun jaringan global di PT. INTI Divisi Manajemen SDM
d) Bagai mana instalasi jaringan *Call Center* di PT. INTI Divisi Manajemen SDM

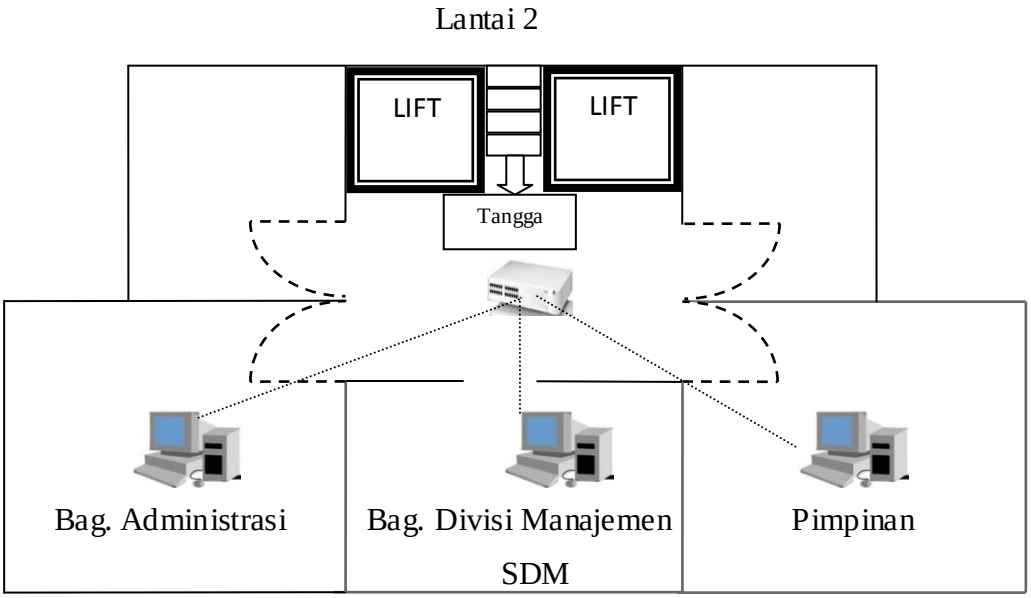
3.2.2 Pemecahan Masalah

Seperti yang telah di jelaskan sebelumnya, bahwa terdapat masalah dalam membangun jaringan lokal dengan denah lokasi pada PT. INTI Divisi Manajemen SDM, untuk itu pemecahan masalahnya yaitu :

1. Topologi jaringan

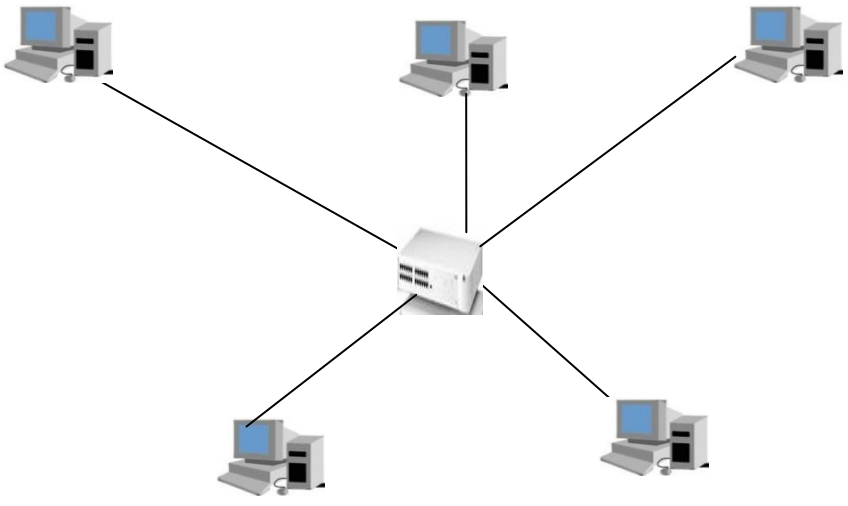
Setelah mengetahui denah bangunan dari PT. INTI Divisi Manajemen SDM, maka topologi yang digunakan yaitu Topologi Star, karna untuk pemasangan kabel dan Hub disarankan mengambil jarak terdekat antara posisi komputer di tiap bidang dalam gedung PT.INTI Divisi Manajemen SDM dengan posisi Hub berada, karena semakin dekat jarak yang diambil maka kebutuhan akan kabel yang dipakai semakin sedikit. Gambar 3.2 Berikut ini adalah gambar Topologi Star yang sudah di sesuaikan dengan kodisi ruangan pada gedung PT. INTI Divisi Manajemen SDM





Gambar 3.2 Topologi Star pada denah gedung PT. INTI Divisi Manajemen SDM

Jaringan *Call Center* pelayanan keluhan pelanggan pada PT. INTI divisi manajemen SDM telah memiliki jaringan yang telah terintegrasi menggunakan jaringan LAN dengan topologi star.



Gambar 3.2 Topologi Star pada PT. INTI di Divisi Manajemen SDM

2. Spesifikasi alat Pembangun jaringan local

a) NIC (Network Interface Card)

Adalah sebuah kartu jaringan yang digunakan untuk menghubungkan komputer dengan media transmisi (kabel). Dalam jaringan pada PT. INTI di Divisi Manajemen SDM. NIC yang digunakan adalah NIC yang telah terdapat pada *Motherboard (onboard).* Gambar 3.3 merupakan contoh NIC (Network Interface Card)



Gambar 3.3 NIC (Network Interface Card)

b) HUB

Hub adalah perangkat jaringan yang terdiri dari banyak port untuk menghubungkan Node atau titik sehingga membentuk jaringan yang saling terhubung dalam topologi star. Hub ada yang mendukung pemggunaan kabel coax yang mendukung topologi BUS dan UTP yang mendukung topologi STAR. Pada PT.INTI Divisi Manajemen SDM di gunakan Hub yang mendukung penggunaan kabel UTP untuk penggunaan topologi STAR. Gambar 3.4 merupakan contoh HUB



Gambar 3.4 HUB

c) Konektor

Untuk menghubungkan kabel UTP diperlukan konektor RJ-45 atau sejenis *jack* yang bentuknya mirip dengan *jack* kabel telepon namun memiliki lebih banyak lubang kabel. Konektor tersebut dipasang di kedua ujung kabel dengan peralatan tang khusus UTP. Gambar 3.5 merupakan contoh Konektor RJ45.



Gambar 3.5 Konektor RJ 45

d) Kabel

Untuk UTP yang dipakai oleh PT.INTI Divisi Manajemen SDM menggunakan UTP kategori 3. Panjang maksimal satu utas kabel LAN adalah 50

meter, bila panjang yang dibutuhkan adalah lebih dari 50 meter. Dengan kabel UTP category 3 dapat mentransmisi data sampai dengan 100Mbps. Gambar 3.6 merupakan contoh kabel UTP.



Gambar 3.6 Kabel UTP

e) Perangkat Lunak

Perangkat lunak yang dipakai pada pembangunan jaringan lokal di PT.INTI Divisi Manajemen SDM, diantaranya :

a. Sistem Operasi Windows XP professional, Windows Server 2003

b. Microsoft Office 2003, Microsoft Office 2007

3. Membangun Jaringan Lokal

Pada saat pembangunan jaringan lokal, hal pertama yang dilakukan adalah mengukur jarak antara komputer pada masing – masing bagian di PT.INTI Divisi Manajemen SDM. Hal ini dilakukan untuk mengatahui seberapa panjang kebutuhan dari kabel yang akan digunakan.

Setelah mengetahui panjang kebutuhan kabel, selanjutnya yaitu mengatur tipe kabel yang digunakan apakah *Straigh*t atau *Cross*. Pada PT.INTI Divisi Manajemen SDM digunakan tipe kabel *Straight* dengan susunan warna kabel seperti pada gambar 3.7 berikut :



<= ujung kiri ujung kanan =>

Gambar 3.7 Susunan kabel *Straight*

Setelah disusun secara *Straight*, lakukan pengetesan dengan *LAN Tester*, seperti pada gambar 3.8. Alat ini berfungsi untuk memeriksa benar tidaknya sambungan kabel. Untuk tipe straight jika benar maka led 1 sampai 8 berkedip.



Gambar 3.8 LAN Tester

Langkah selanjutnya yaitu mengatur IP masing – masing dari komputer yang digunakan dan terhubung ke dalam jaringan lokal, caranya yaitu Start Menu – Control Panel – Network Connection – Local Area Connection – Internet Protocol (TCP / IP) – Propertis, selanjutnya pilih Use The Following IP Address dan masukan IP yang akan dipakai :

IP Address 1	: 192.168.0.2	IP pada komputer Pimpinan
Subnet Mask	: 255.255.255.0	
Default Gateway	: 192.168.0.1	

IP Address 1	: 192.168.0.3	IP pada komputer
Subnet Mask	: 255.255.255.0	
Default Gateway	: 192.168.0.1	Divisi Manajemen SDM

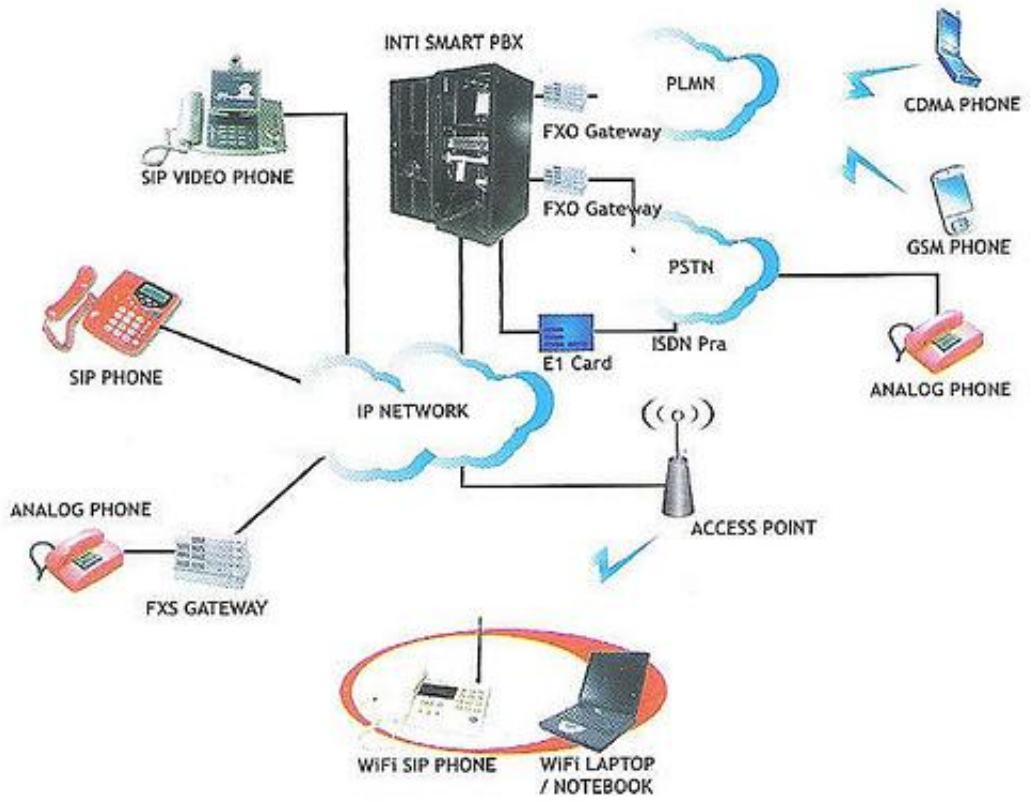
IP Address 1	: 192.168.0.4	IP pada komputer
Subnet Mask	: 255.255.255.0	
Default Gateway	: 192.168.0.1	Bag. Pemasaran

IP Address 1	: 192.168.0.5	IP pada komputer Bag. Administrasi
Subnet Mask	: 255.255.255.0	
Default Gateway	: 192.168.0.1	

IP Address 1	: 192.168.0.6	IP pada komputer Bag. Gudang
Subnet Mask	: 255.255.255.0	
Default Gateway	: 192.168.0.1	

Setelah semua IP di atur, lalu atur grup untuk jaringan yang telah di buat. Caranya yaitu klik kanan pada My Computer – Propertis – Computer Name – Change – Work Group. Nama work group harus sama dengan semua komputer yang ada.

Berdasarkan perkembangan selanjutnya, Divisi manajemen SDM membutuhkan sebuah jaringan *Call center* yang bermaksud untuk menangani keluhan *customer*. Adapun gambar denah jaringan *call center* yang ada di PT.INTI Divisi Manajemen SDM adalah sebagai berikut :



Gambar 3.9 Jaringan *Call Center* PT.INTI Divisi Manajemen SDM

3.2.3 Topologi yang digunakan dalam jaringan *Call Center*

Biasanya *user* memanfaatkan jaringan untuk berbagi sumberdaya seperti printer, dan *sharing data*. Untuk Internet, *user* menggunakannya sebagai akses informasi dan komunikasi (email). Dalam kasus ini kita akan membangun jaringan *call center* . Dalam membangun jaringan *Call Center* di PT.INTI Juga kita memerlukan Perancagan jaringan mulai dari kebutuhan *hardware* dan kebutuhan *Software*.

a. Dalam Jaringan *Call center* yang terdapat di PT.INTI divisi manajemen SDM menggunakan topologi star karena terdapat konsentrator yaitu switch dimana didalamnya terdapat ADMIN dan *Customer Service*.
b. Dalam jaringan *Call Center* terdapat ADMIN dan *Customer Service,* dimana sebuah jaringan *Call Center* membutuhkan konsentrator yaitu switch, *Call Center* merupakan sebuah jaringan besar yang memerlukan 2 buah switch atau bahkan lebih. Untuk menghubungkan setiap switch ke switch yang lainnya digunakan topologi bus.
c. Topologi *hybrid,* gabungan antara topologi bus dan topologi star terbentuk yaitu pada LAN ADMIN dan LAN *Customer service*. Faktor gedung dan jumlah lantai yang ada mempengaruhi terbentuknya topologi *hybrid* ini.

Dengan demikian topologi jaringan yang digunakan dalam pembangunan jaringan *Call Center* di PT.INTI tersebut, antara lain :

1. Topologi *Bus,*
2. Topologi *Star,* dan
3. Topologi *hybrid*

3.2.4 Kebutuhan Perangkat keras Jaringan *Call Center*

Penggunaan sistem komputer sebagai alat bantu dalam menyelesaikan tugas-tugas atau pekerjaan sudah bukan menjadi hal yang aneh, tapi merupakan suatu keharusan karena banyak kemudahan-kemudahan yang bisa diperoleh Komputer terdiri perangkat keras dan perangkat lunak. Perangkat lunak

memberikan instruksi-instruksi kepada perangkat keras untuk melakukan suatu tugas tertentu.

Penggunaan komputer sebagai alat bantu suatu kejadian yang benar-benar terjadi dikehidupan nyata yang sering digunakan, oleh karena itu penyusun berusaha untuk membuat salah satu peristiwa atau kejadian yang terjadi didunia nyata yaitu Jaringan *Call Center* dengan menggunakan komputer.

Tabel 3.1 Perangkat Keras yang Dibutuhkan untuk Jaringan *Call Center* di PT. INTI Divisi Manajemen SDM

No	Perangkat Keras	Volume	Spesifikasi	Keterangan
1	Modem ADSL	1 Buah	Kecepatan 2 Mbps	Untuk koneksi ke ISP disarankan menggunakan Speedy.
2	server	2 Buah	• Spesifikasi IBM XSeries Xeon Dual Core, 2.0 Ghz, Memory 2 GB, Hardisk 250 Gb • Gateway server (bisa komputer biasa1atau server branded2)	Gateway server digunakan sebagai *Network Address Translation* (NAT) dan *router* atau sebagai proxy. Database server memerlukan respon yang cepat karena akan terinstall suatu aplikasi, sehingga memerlukan kerja prosessor, memory dan hardisk yang besar.
3	Hub/Switch	Lebih dari 15 switch		Untuk 4 grup LAN
4	Router	1 Buah	Server IBM, P4 Xeon, 2.0 Ghz, Memory 1 Ghz, Hardisk 73 GB	Server yang difungsikan sebagai Router biasa disebut PC Router, dapat juga membeli mesin Router yang sudah memilki aplikasi.
5	Kabel	secukupnya	Kabel UTP	

3.2.5 Kebutuhan Perangkat Lunak Jaringan *Call Center*

Software memiliki dua peran, di satu sisi berfungsi sebagai sebuah produk dan di sisi lain sebagai pengontrol pembuatan sebuah produk. Sebagai produk, *software* mengantarkan potensi perhitungan yang dibangun oleh software komputer. *Software* merupakan transformer informasi yang memproduksi, mengatur, memperoleh, memodifikasi, menampilkan atau memancarkan informasi, ini dapat sesederhana suatu bit tunggal atau sekompleks sebuah simulasi multimedia. Sedangkan peran sebagai pengontrol yang dipakai untuk mengantarkan produk, *software* berlaku sebagai dasar untuk kontrol komputer (sistem operasi), komunikasi informasi (jaringan), dan penciptaan serta kontrol dari program-program lain (peranti dan lingkungan *software*). *Software* yang sedang berjalan dalam jaringan *Call Center* di PT.INTI tersebut antara lain *Vicidial*, OTRS, *Asterisk* dan *LinuxSuse*.

3.3 Implementasi Jaringan *Call Center*

Pada bagian ini akan dijelaskan implementasi pada Jaringan *call center* di PT INTI divisi manajemen SDM, dalam implementasi ini akan di jelaskan cara meng*install* dan mengkonfigurasikan *software* OTRS, *Vicidial* dan *Asterisk*.

3.3.1 Menginstal database OTRS secara manual

Jika Anda tidak dapat menggunakan web installer untuk mengeset database OTRS, Anda dapat mengeset secara manual. Script dengan statmen SQL

untuk membuat dan mengkonfigurasi database yang berada di scripts/database pada home directory OTRS user.

```
linux:~# cd /opt/otrs/scripts/database/
linux:/opt/otrs/scripts/database# ls
otrs-initial_insert.db2.sql        otrs-schema.mysql.sql
otrs-schema.oracle.sql
otrs-initial_insert.mssql.sql      otrs-schema-post.db2.sql
otrs-initial_insert.mysql.sql      otrs-schema.postgresql.sql
otrs-initial_insert.oracle.sql
otrs-initial_insert.postgresql.sql  otrs-schema-post.mssql.sql
otrs-initial_insert.xml            otrs-schema-post.mysql.sql
otrs-schema.db2.sql                otrs-schema-post.oracle.sql
 otrs-schema-post.postgresql.sql
otrs-schema.mssql.sql              otrs-schema.xml
linux:/opt/otrs/scripts/database#
```

Untuk mengeset database untuk database back-ends yang berbeda, file .sql harus diproses dengan perintah khusus.

Langkah – langkah membuat OTRS database secara manual

1. Membuat DB : membuat database, yang Anda inginkan untuk digunakan untuk OTRS, dengan *Clientt* database Anda atau interface database Anda.

2. Membuat table: dengan otrs-schema.DatabaseType.sql files (misal otrs-schema.oracle.sql, otrs-schema.postgresql.sql) Anda dapat mmembuat table pada OTRS database Anda.
3. Memasukan inisial system data: OTRS membutuhkan beberapa inisial system data untuk bekerja dengan baik (misal. the different ticket states, ticket and notification types). Tergantung pada tipe database Anda. Gunakan salah satu file otrs-initial_insert.mysql.sql, otrs-initial_insert.db2.sql, otrs-initial_insert.oracle.sql, otrs-initial_insert.postgresql.sql atau otrs-initial_insert.mssql.sql .
4. Membuat referensi antara table: langkah terakhir yaitu membuat referensi antara table yang berbeda pada OTRS database. Gunakan otrs-schema post.DatabaseType.sql files untuk membuat referensi ini (misal otrs-schema-oracle.post.sql, otrs-schema-post.postgresql.sql).

Setelah selesai mengeset database, Anda harus mengecek dan set hak akses untuk OTRS database. Harus memadai untuk memberikan akses ke satu pengguna. Tergantung pada database *Server* Anda menggunakan pengaturan hak akses yang berbeda, tapi harus memungkinkan baik dengan database *Clientt* Anda atau graphical database front-end Anda.

Jika database Anda dan hak akses sudah terkonfigurasi dengan baik, Anda harus memberitahu OTRS database back-end yang Anda ingin gunakan dan bagaimana ticket system dapat terkoneksi ke database. Buka file

Kernel/Config.pm pada home directory OTRS user dan ubah parameter berikut sesuai dengan kebutuhan Anda

```
# DatabaseHost
# (The database host.)
$Self->{'DatabaseHost'} = 'localhost';
# Database
# (The database name.)
$Self->{Database} = 'otrs';
# DatabaseUser
# (The database user.)
$Self->{DatabaseUser} = 'otrs';
# DatabasePw
# (The password of database user.)
$Self->{DatabasePw} = 'some-pass';
```

3.3.2 Seting cron jobs untuk OTRS

OTRS membutuhkan beberapa cron jobs untuk bekerja dengan baik. Cron jobs harus berjalan dengan hak user yang sama yang telah ditentukan untuk modul OTRS. Itu berarti cron jobs harus dimasukkan ke dalam file crontab dari OTRS user. Semua script cron jobs berada di `var/cron` pada home directory OTRS user.

```
linux:~# cd /opt/otrs/var/cron

linux:/opt/otrs/var/cron# ls

aaa_base.dist              pending_jobs.dist        session.dist

fetchmail.dist             postmaster.dist          unlock.dist

generic_agent-database.dist  postmaster_pop3.dist

generic_agent.dist         rebuild_ticket_index.dist


linux:/opt/otrs/var/cron#
```

Semua script berhenti di .dist . Anda harus menyalinnya ke file yang tidak berakhir. Jika Anda menggunakan bash, Anda gunakan perintah di bawah ini..

```
linux:/opt/otrs/var/cron# for foo in *.dist; do cp $foo

`basename $foo .dist`; done

linux:/opt/otrs/var/cron# ls

aaa_base                   generic_agent.dist    rebuild_ticket_index

aaa_base.dist              pending_jobs

rebuild_ticket_index.dist

fetchmail                  pending_jobs.dist     session

fetchmail.dist             postmaster            session.dist

generic_agent              postmaster.dist       unlock

generic_agent-database     postmaster_pop3       unlock.dist

generic_agent-database.dist  postmaster_pop3.dist


linux:/opt/otrs/var/cron#
```

Table berikut menjelaskan beberapa script yang bekerja dan mengapa mereka dibutuhkan untuk cron job untuk OTRS user.

Table 3-2. Penjelasan cron job scripts

Script	**Function**
aaa_base	Script ini mengeset dasar crontab OTRS user
fetchmail	Jika mail baru akan diambil dengan fetchmail ke dalam ticket system, script ini bisa digunakan
generic_agent	Script ini menjalankan pekerjaan GenericAgent harus diambil dan tidak disimpan dalam database, tapi di file konfigurasi sendiri.
generic_agent-database	Script ini menjalankan pekerjaan GenericAgent tetapi tidak di simpan dalam database.
pending_jobs	Script ini mengecek system untuk menunggu (pending) tickets.
postmaster	Script ini mengecek antrian pesan dari ticket system dan mengirimkan pesan yang masih dalam antrian.
postmaster_pop3	Script ini menjemput mails dari akun POP3 yang ditentukan admin area pada bagian "PostMaster POP3 Account".
rebuild_ticket_index	Script ini membangun kembali ticket index. Meningkatkan kecepatan QueueView.
session	Script ini menghapus yang sudah tidak diperlukan ID's
unlock	Script ini membuka ticket dalam system.

Untuk mengeset semua cron jobs, script `bin/Cron.sh` dapat digunakan, yang berada di home directory OTRS user. Script ini memerlukan sebuah parameter ketika dijalankan yang memberitahukan jika Anda ingin utnuk di install, hapus atau reinstall cron jobs. Gunakan parameter di bawah ini :

Cron.sh { *start* } { *stop* } { *restart* } [*OTRS user*]

Karena cron jobs membutuhkan install file crontab OTRS user, Anda perlu masuk sebagai OTRS user. Jika Anda masuk sebagai root, Anda dapat merubah OTRS user dengan perintah **su otrs**. Jalankan perintah install cronjobs di bawah ini :

```
linux:/opt/otrs/var/cron# cd /opt/otrs/bin/
linux:/opt/otrs/bin# su otrs
linux:~/bin$ ./Cron.sh start
/opt/otrs/bin
Cron.sh - start/stop OTRS cronjobs - <$Revision: 1.28 $>
Copyright (C) 2001-2009 OTRS AG, http://otrs.org/
(using /opt/otrs) done
linux:~/bin$ exit
exit
linux:/opt/otrs/bin#
```

3.3.3 Menginstall *Vicidial* secara manual

Sebelum menginstall *Asterisk* terlebih dahulu kita menginstall *Vicidial* Pada tahap ini kita akan menginstal perangkat lunak yang *Asterisk* dan / atau kebutuhan astGUI*Clientt* untuk dapat berfungsi kemampuannya sepenuhnya. Tidak semua paket perangkat lunak sudah manditory berhasil menginstal *Asterisk* dan beberapa dari mereka dapat diinstal pada komputer lain pada jaringan Anda seperti MySQL atau Apache. Tapi, dalam hal ini instalasi kita mengasumsikan bahwa tidak ada komputer lain di jaringan kami untuk membantu *Asterisk Server*, sehingga harus memiliki segala sesuatu yang perlu diinstal secara lokal.

PERL menginstal modul:

ketik di shell linux

```
perl -MCPAN -e shell
```

Kemudian akan melalui setup CPAN, ketik 'tidak' ketika prompt 'Apakah Anda siap untuk konfigurasi manual? " Setelah melihat CPAN kita dapat mulai menginstal modul, Jika belum pernah menginstal CPAN maka harus menginstal modul pertama berikut ini :

```
install MD5

install Digest::MD5

install Digest::SHA1

install readline
```

```
install Bundle::CPAN

reload cpan

install DBI

force install DBD::mysql

install Net::Telnet

install Time::HiRes

install Net::Server

install Unicode::Map

install Jcode

install Spreadsheet::WriteExcel

install OLE::Storage_Lite

install Proc::ProcessTable

install IO::Scalar

install Spreadsheet::ParseExcel
```

Berhenti dan selesai menginstall CPAN.

3.3.4 Instalasi *Asterisk*

ikuti langkah-langkah baris perintah:

```
mkdir /usr/src/Asterisk

cd /usr/src/Asterisk

wget http://ftp.digium.com/pub/Asterisk/releases/Asterisk-1.2.17.tar.gz

wget http://ftp.digium.com/pub/zaptel/releases/zaptel-1.2.16.tar.gz

wget http://ftp.digium.com/pub/libpri/releases/libpri-1.2.4.tar.gz

gunzip Asterisk-1.2.17.tar.gz

tar xvf Asterisk-1.2.17.tar

gunzip zaptel-1.2.16.tar.gz

tar xvf zaptel-1.2.16.tar

gunzip libpri-1.2.4.tar.gz

tar xvf libpri-1.2.4.tar
```

Zaptel mengkompilasi dan Libpri menggunakan perintah berikut

```
cd ./zaptel-1.2.16

make clean

make
```

```
make install

cd ../libpri-1.2.4

make clean

make

make install
```

Kompilasi *Asterisk*, mengambil langkah-langkah berikut. (Catatan: ketika menjalankan "patch" perintah itu akan meminta untuk memasukkan file yang ingin patch, Anda akan perlu input secara manual nama file yang sesuai, tidak hanya secara membuta *copy & paste* kode di bawah ini, misalnya dalam pertama yang akan Anda app_meetme.c patch dan patch kedua akan diterapkan pada cli.c)

```
cd ../Asterisk-1.2.17

cd apps

wget http://www.eflo.net/files/meetme_volume_control_1.2.16.patch

patch -p1 < ./meetme_volume_control_1.2.16.patch

- File to patch: app_meetme.c

cd ../

wget http://www.eflo.net/files/cli_chan_concise_delimiter.patch

patch -p1 < ./cli_chan_concise_delimiter.patch
```

```
- File to patch: cli.c

make clean

make

make install

make samples
```

Sekarang instalasi *Asterisk* telah dibangun, dimuat dan sudah bisa untuk mengkonfigurasinya.

Pemasangan Astgui*Clientt* dan *Vicidial*:

Sekarang, *Asterisk* terinstal dan dijalankan kita dapat menambahkan komponen astGUI*Clientt* dan *VICIDIAL* untuk sistem. Pergi ke http://astgui*Clientt*.sf.net/ dan mendownload paket astgui*Clientt* terbaru (seperti tulisan ini adalah 2.0.3), dan ikuti langkah-langkah;

```
mkdir /usr/src/astgui*Clientt*

cd /usr/src/astgui*Clientt*

wget

http://internap.dl.sourceforge.net/sourceforge/astgui*Clientt*/astgui*Clientt*_2.0.3.zip

unzip astgui*Clientt*_2.0.3.zip

perl install.pl
```

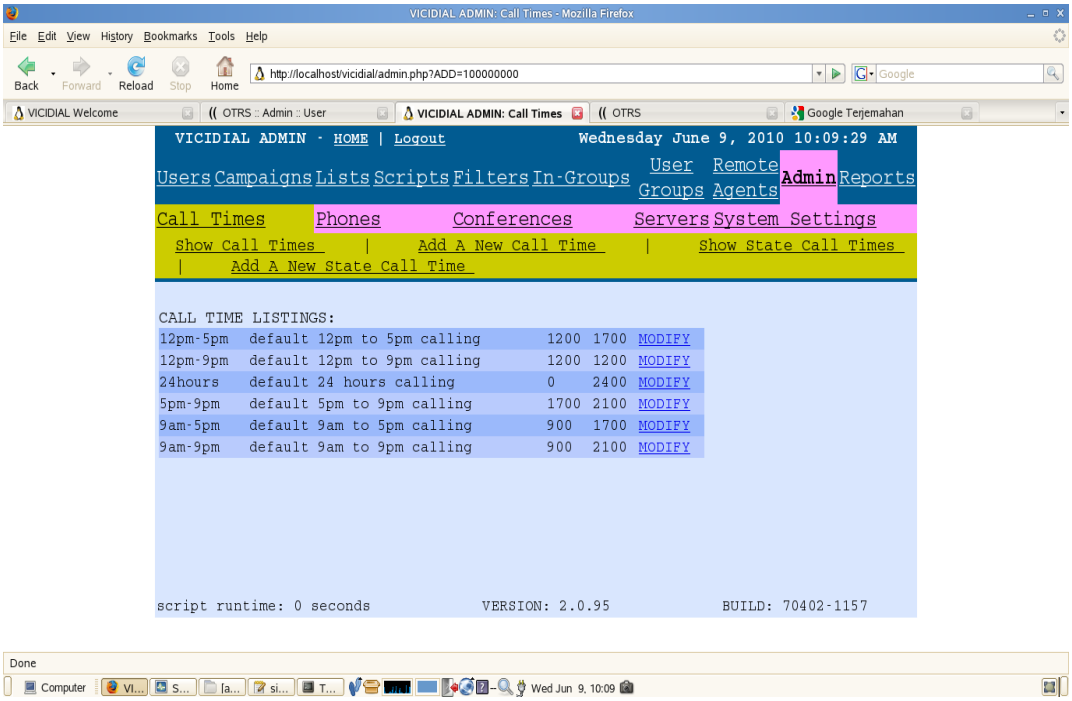
3.3.5 Perancangan Antar Muka Jaringan *Call Center*

3.3.5.1 *VICIDIAL*

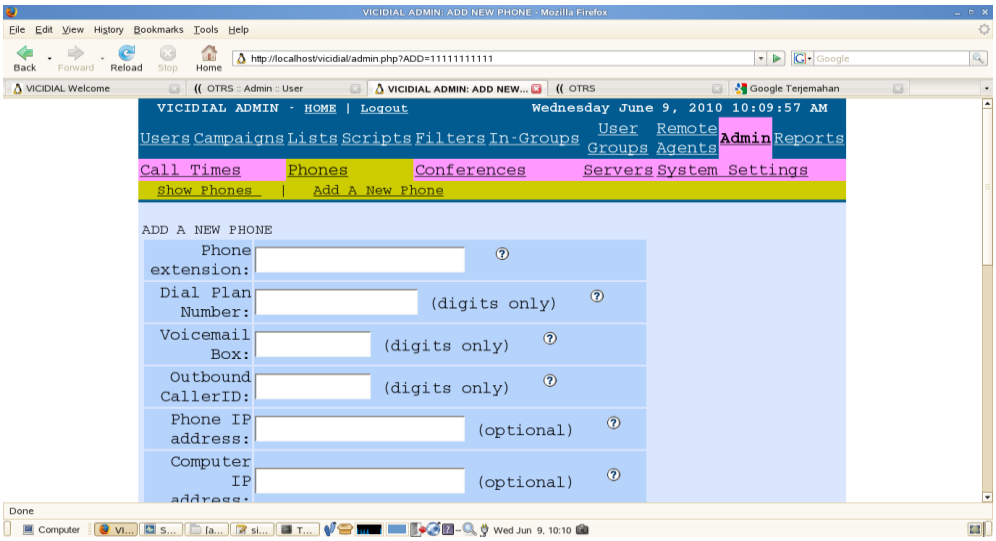
Vicidial instalasi, konfigurasi, kustomisasi dan dukungan *Vicidial* adalah fungsional penuh outbound dan inbound *Call Center* solusi. *Vicidial* adalah paket program-program yang dirancang untuk berinteraksi dengan sistem Open-Source *Asterisk* PBX Telepon pada tingkat komputer klien untuk memperluas fungsionalitas dari telepon Kita dan sistem.

1. Tampilan Antar Muka Admin *VICIDIAL*.

Adapun tampilan antar muka Admin *VICIDIAL* di PT Inti adalah sebagai berikut :



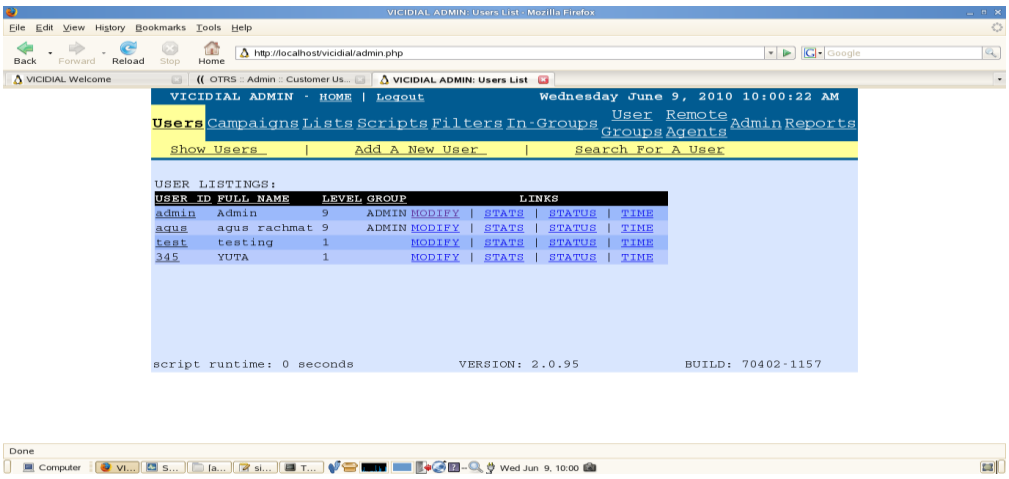
Gambar 3.10 *Tampilan Admin Call Times*



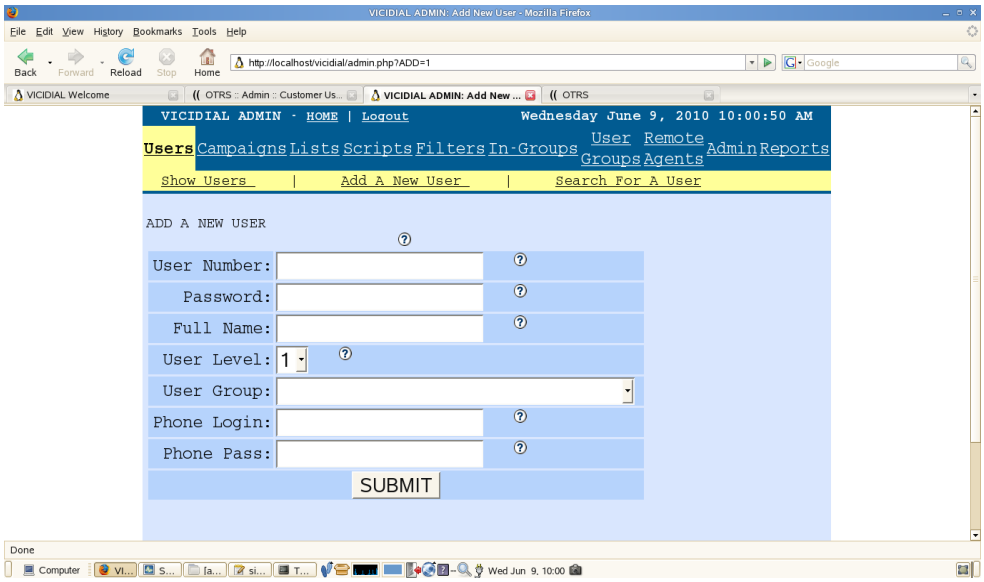
Gambar 3.11 *Tampilan Admin Phone*

2. Tampilan Antar Muka Users *VICIDIAL*

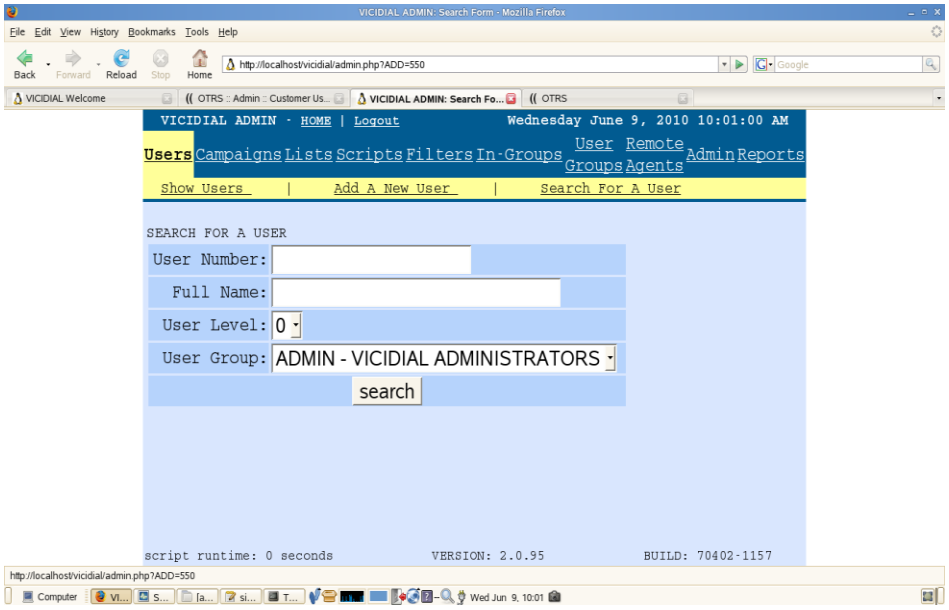
Adapun tampilan antar muka Users *VICIDIAL* di PT Inti adalah adalah sebagai berikut :



Gambar 3.12 *Tampilan User (Show Users)*



Gambar 3.13 *Tampilan Antar Muka Users (Add A New Users)*

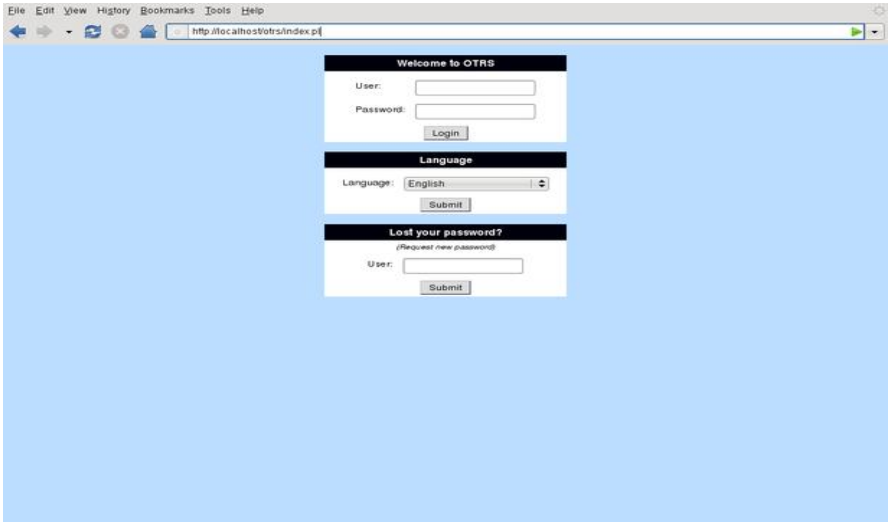


Gambar 3.14 *Tampilan Antar Muka Users (Search For A User)*

3.3.6 OTRS

Open Ticket Request System (OTRS) adalah sebuah aplikasi web yang dapat digunakan dengan setiap web browser HTML yang kompatibel. Untuk Memulai sistem OTRS pastikan tidak ada aplikasi *Server* lain yang berjalan pada port 25 (smtp), 80 (http), 443 (https), 3306 (MySQL). Jika demikian, cari tahu

apakah masih ada aplikasi *Server* lain yang masih berjalan di mesin komputer. Contoh adalah Microsoft IIS web*Server* berjalan tanpa suara di latar belakang atau layanan SMTP corresponding. Sebelum memulai OTRS tutup *Server* apikasi yang masih terbuka.



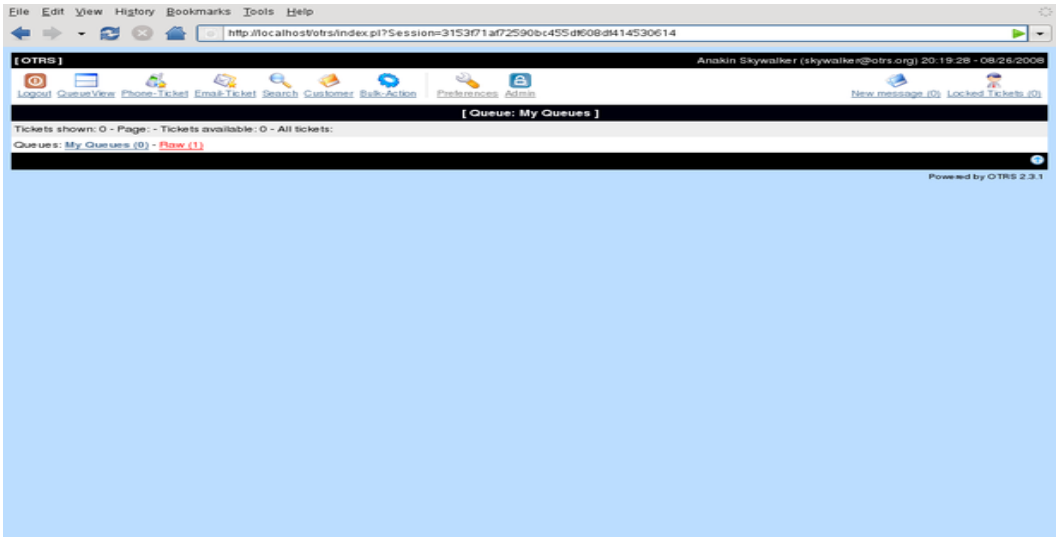
Gambar 3.15 *Tampilan pada saat Login*

Pada layar, Anda dapat memasukkan nama pengguna dan sandi. Karena tidak ada pengguna dibuat setelah instalasi sistem baru, Anda harus login sebagai administrator OTRS pertama. Untuk login sebagai admin menggunakan OTRS "root @ localhost" untuk nama pengguna dan "root" untuk password.

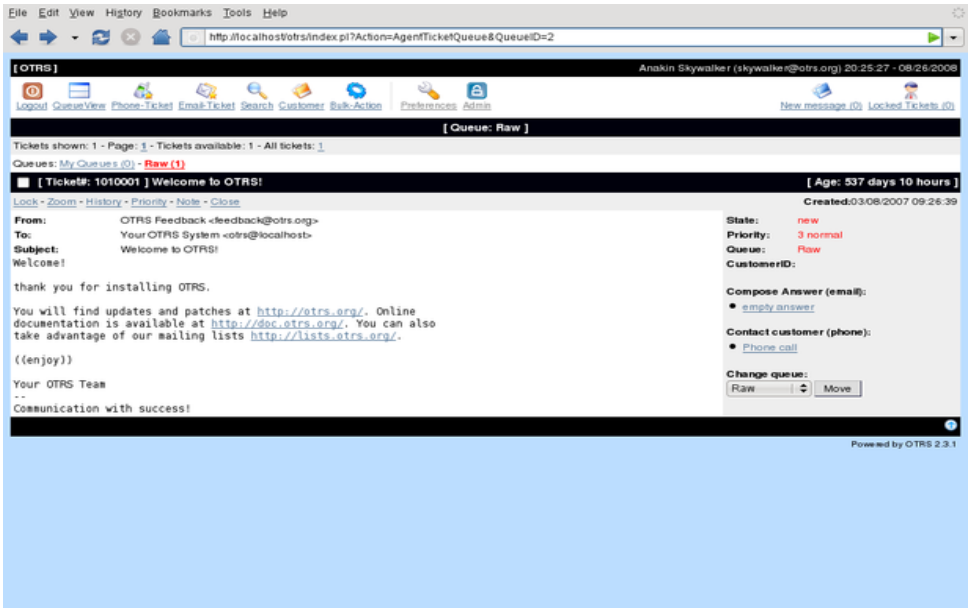
Jika Anda tidak ingin login sebagai administrator OTRS cukup masukkan nama pengguna dan sandi untuk account agen normal Anda. Dengan daftar kotak di bawah field input untuk nama pengguna dan sandi Anda dapat memilih bahasa yang ingin Anda gunakan pada antarmuka web.

Jika Anda lupa sandi Anda, Anda bisa mendapatkan password baru melalui sistem. Cukup masukkan alamat email yang terdaftar untuk Anda OTRS account ke dalam kolom input di bagian bawah layar login dan mengirimkan input.4.5. An overview to the web interface

Setelah Anda berhasil login ke dalam sistem antarmuka web dimuat. default Per Anda berada di QueueView setelah login. QueueView menyediakan sebuah gambaran singkat tentang tiket di antrian yang berbeda, memberitahukan tentang pesan baru, memberitahu Anda berapa banyak mengunci tiket yang Anda miliki, misalnya

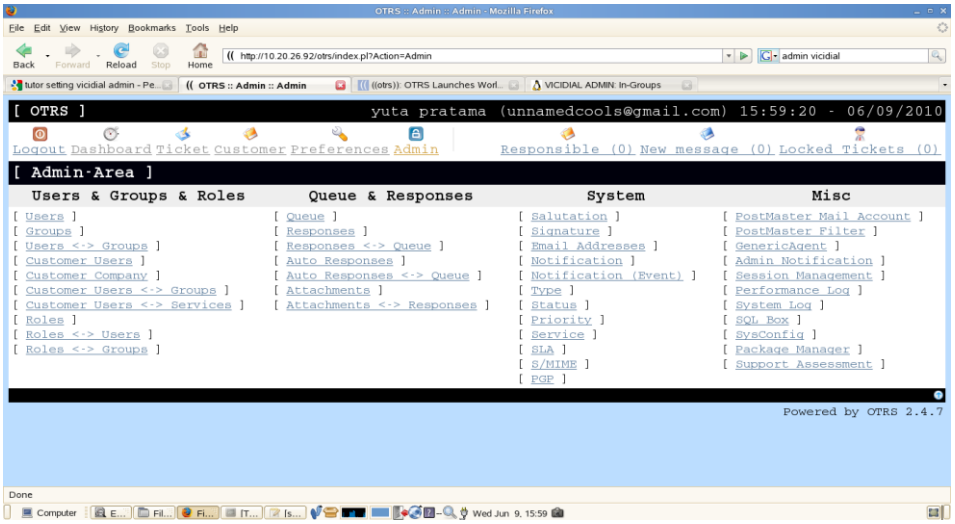


Gambar 3.16 *Tampilan Queue / Tiket Antrian (1)*

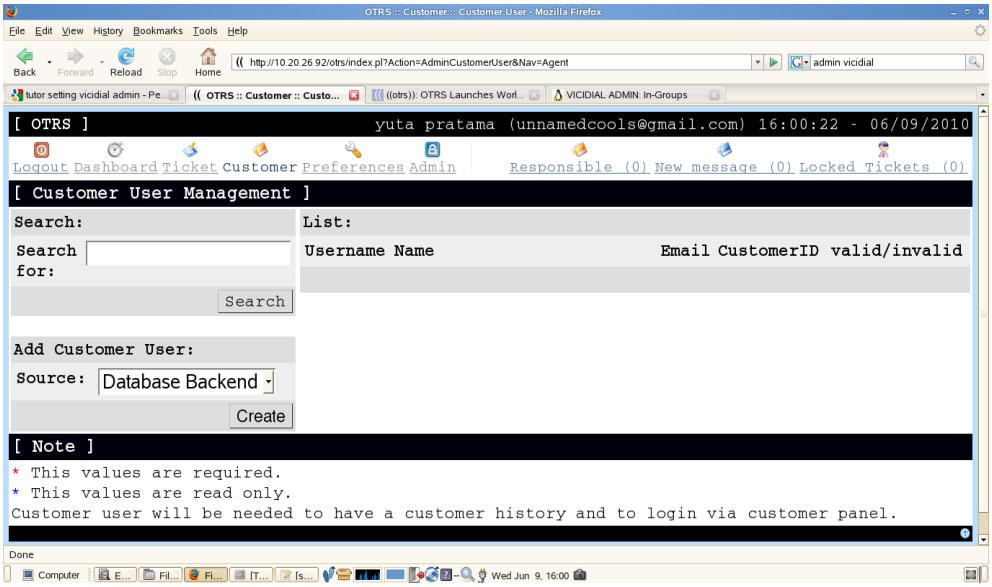


Gambar 3.17 *Tampilan Queue / Tiket Antrian (2)*

Adapun tampilan antar muka Admin adalah sebagai berikut :



Gambar 3.18 *Tampilan Antar Muka OTRS (Admin - Area)*

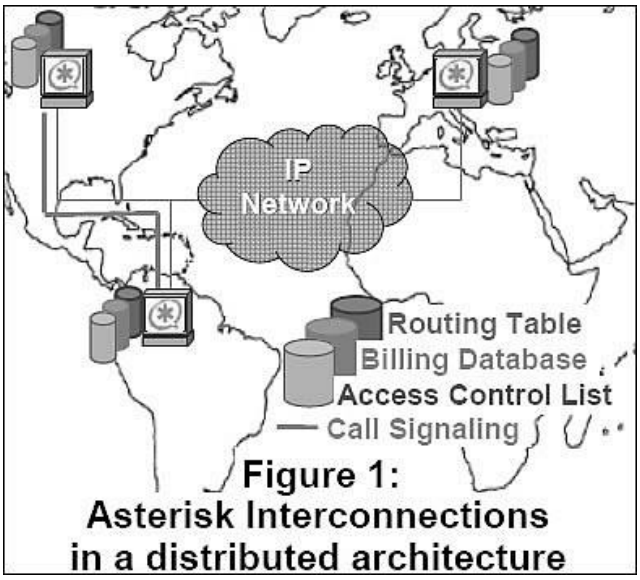


Gambar 3.19 *Tampilan Antar Muka OTRS (Customer User Management)*

3.3.7 *ASTERISK*

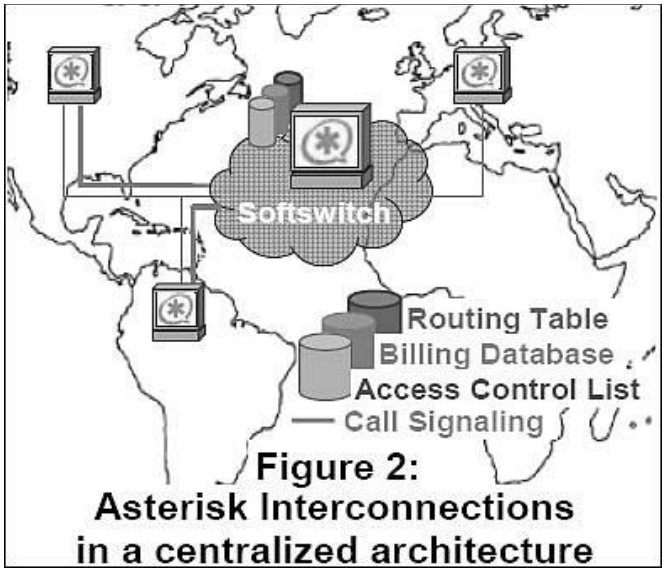
Asterisk sering disebut sebagai Dan "PBX open source". Kita dapat menggunakan *Asterisk* untuk membangun sebuah PBX, PBX / *PABX* (*Private Automatic Brance Exchange*) merupakan suatu teknologi komunikasi yang mengatur hubungan *telephone* antar pelanggan tanpa harus melalui sentral lokal, serta berfungsi sebagai*gateway* dalam menghubungkan ke jaringan *PSTN*. *Switch / Router*berfungsi mengarahkan paket data yang datang ke jaringan data sesuai dengan alamat tujuannya. Secara konvensional terdapat 2 jaringan yang berbeda yaitu jaringan suara (*Circuit Network*) dan jaringan paket data (*Packet Data Network*). Akan tetapi seiring dengan perkembangan teknologi telekomunikasi, terdapat suatu teknologi yang memungkinkan komunikasi suara dan faksimili mengalami proses paketisasi dan dikirimkan melalui jaringan paket data yang dikenal dengan teknologi *Voice over Internet Protocol (VoIP).* PBX hanayalah

satu dari banyak aplikasi yang dapat dibangun dengan *Asterisk*. Gambaran lain dari *Asterisk* :



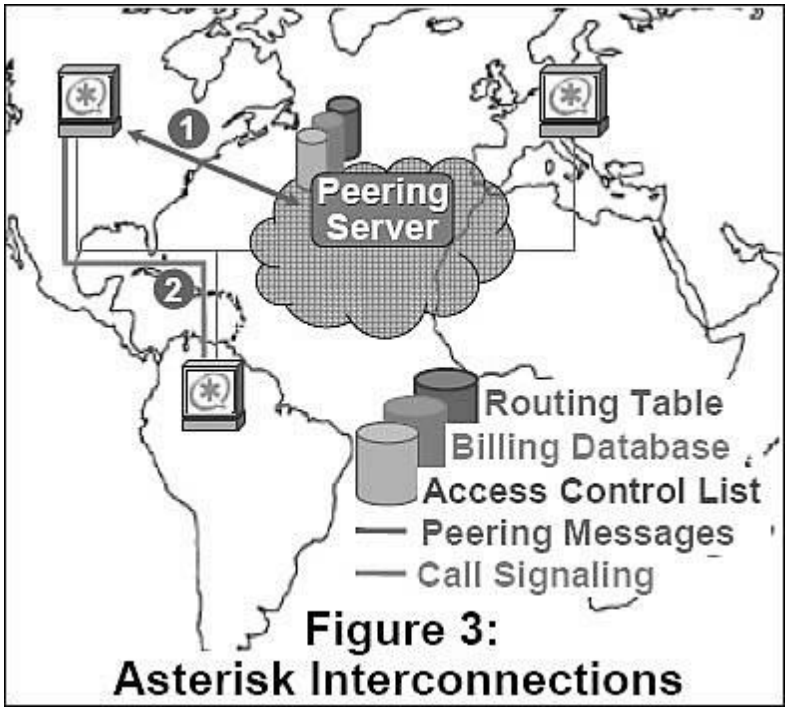
Gambar 3.20 *Tampilan Asterisk Interconnections*

in a distributed Architecture (Figure 1)



Gambar 3.21 *Tampilan Asterisk Interconnections*

In a centralized Architecture (Figure 2)



Gambar 3.22 *Tampilan Asterisk Interconnections (Figure 3)*