

SKRIPSI

PENGEMBANGAN APLIKASI BILLING SYSTEM BERBASIS *OPEN SOURCE* di RT/RW NET



Disusun Oleh

RAHADIAN INDRA NUGROHO

04.12.616



**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2010**

LEMBAR PERSETUJUAN

PENGEMBANGAN APLIKASI BILLING SYSTEM BERBASIS
OPEN SOURCE di RT/RW NET

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Komputer dan Informatika Strata Satu (S-1)*

Disusun Oleh :

RAHADIAN INDRA NUGROHO

NIM : 04.12.616

Diperiksa dan Disetujui

Dosen Pembimbing I

Dosen Pembimbing II



Ir. TH. Mimien Mustikawati, MT

NIP.1030000352

Sotyohadi ST.

NIP.Y.1039700309

Mengetahui,

Ketua Jurusan Teknik Elektro S-1



Ir. F. Yudi Limpraptono, MT

NIP.Y 103 950 0274

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

2010

ABSTRAK

Nama : Rahadian Indra Nugroho, NIM : 04.12.616, Jurusan : Teknik Komputer dan Informatika S-1, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang, Judul Skripsi : " PENGEMBANGAN APLIKASI BILLING SYSTEM BERBASIS OPEN SOURCE di RT/RW NET"

**Dosen Pembimbing : I Ir. Mimien Mustkawati, MT
II Sotyohadi ST**

Pertemuan bisnis yang memerlukan koneksi internet dapat dilakukan tidak terbatas di perkantoran saja, tetapi dapat dilakukan di semua public area seperti warnet, rumah, atau area hotspot yang telah banyak tersedia. RT/RW Net inilah solusi atau alternatif untuk mengakses Internet bagi rumah tangga dan mengatasi mahal dan langkahnya sambungan telepon. Seiring dengan perkembangan teknologi yang demikian pesatnya maka penggunaan Hotspot untuk kepentingan umum khususnya di RT/RW Net. Oleh karena itu seiring dengan perkembangan teknologi yang demikian pesatnya maka dibutuhkan sebuah media yang dapat mengelola penggunaan Hotspot untuk kepentingan umum khususnya di RT/RW Net yang berbasis web.

Dalam aplikasi ini menunjukkan bahwa dengan adanya Manajemen Hotspot ini mempermudah proses pengelolaan jaringan internet khususnya penggunaan wireless dan mengurangi biaya pengelolaan.

Katakunci : Wireless, Billing System, RT/RW.net, Hostpot, DAD, Linux, Aphase, MySQL

KATA PENGANTAR

Alhamdulillah, segala puji hanya milik Allah SWT. Dialah pencipta alam semesta dan seisinya. Zat yang Maha Mengetahu segala sesuatu. karena dengan segala rahmat dan hidayah-Nya, telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul :

“ PENGEMBANGAN APLIKASI BILLING SYSTEM BERBASIS *OPEN SOURCE* di RT/RW NET“.

Pembuatan skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata I di Institut Teknologi Nasional Malang. Dalam penyusunan skripsi ini penulis banyak mendapat bantuan baik moril maupun materiil, saran dan dorongan semangat dari berbagai pihak, untuk itu penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi MSEE., selaku rektor ITN Malang
2. Bapak Ir. Sidik Noetjahjono, MT selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang
3. Bapak Ir. F. Yudi Limpraptono MT., selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
4. Ibu Ir. TH Mimien Mustikawati MT, selaku Dosen Pembimbing I.
5. Bapak Sotyo Hadi ST, selaku Dosen Pembimbing II.
6. Kedua orang tua yang telah memberikan dukungan untuk selalu berdoa dan berusaha.

7. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan.

Akhir kata, penulis mohon maaf kepada semua pihak bilamana selama penyusunan skripsi ini penulis membuat kesalahan baik yang disengaja maupun yang tidak sengaja dan semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, Februari 2010

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAKSI	iii
KATA PENGANTAR.....	iv
DAFTAR ISI	vi
DAFTAR GAMBAR.....	x
DAFTAR TABEL	xiii
BAB I : PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan.....	3
1.5 Metodologi	3
1.6 Sistematika Penulisan.....	5
BAB II : TINJAUAN PUSTAKA	6
2.1 Pengertian Billing.....	6
2.2 Pengertian Sistem (System).....	6
2.2.1 Pemodelan Sistem.....	7
2.2.2 Rancangan Basis Data (Database).....	11
2.2.3 Rancangan Input	11
2.2.4 Rancangan Output	12
2.3 Pengertian Billing System (Sistem Billing).....	12

2.4	Pengertian Internet.....	12
2.4.1	World Wide Web	14
2.5	Pengertian Open Source	15
2.6	Perangkat Lunak Yang digunakan	17
2.6.1	Apache	17
2.6.2	PHP (PHP Hypertext Preprocesor)	18
2.6.3	My SQL	20
2.6.4	FreeRadius.....	21
2.6.5	Chillispot	31
BAB III	: PERANCANGAN DAN DESAIN SISTEM	33
3.1	Deskripsi Objek Penelitian	33
3.1.1	Sejarah Singkat RT/RW NET	33
3.1.2	Implementasi Awal RT/RW-net	33
3.2	Hasil Penelitian dan Pembahasan	34
3.2.1	Hasil Penelitian	34
3.2.2	Perancangan Sistem	35
3.2.3	Bagan Alir <i>Billing System</i>	37
3.2.4	Diagram Arus Data.....	46
3.2.5	Perancangan Data Base.....	50
3.2.6	Desain Billing System	59
3.2.7	Desain Input.....	65
3.2.8	Desain Output.....	70
BAB IV	: IMPLEMENTASI DAN PENGUJIAN HASIL	73
4.1	Implementasi	73

4.1.1 Instalasi OS (<i>Operating System</i>).....	73
4.1.2 Persiapan <i>Hardware Server</i>	73
4.1.3 <i>Install ubuntu</i>	74
4.1.4 Proses <i>Remake</i> dari Ubuntu ke Distro RT/RW NET	
Hostpot	74
4.1.4.1 Software Reqrutment.....	74
4.1.4.2 System Configurasi.....	75
4.1.4.3 Program dan Support Instalation	77
4.1.4.4 Remake with Remastering and Reconstructor	91
4.1.4.5 Remastersys Backup.....	91
4.1.4.6 Rekonstrutor.....	98
4.1.5 Halaman <i>add certificate</i> RT/RW NET Hotspot	102
4.2 Pengujian System	103
4.2.1 Halaman Login User.....	103
4.2.2 Halaman Home Control Panel Chasier.....	104
4.2.3 Halaman Postpaid Account Management.....	105
4.2.4 Halaman Voucher Management	106
4.2.5 Halaman Invoice Management.....	107
4.2.6 Halaman Statistic.....	108
4.2.7 Halaman Chasier Change Pasword.....	109
4.2.8 Halaman Home Control Panel Adminstratir	110
4.2.9 Halaman Cashier Management.....	111
4.2.10 Halaman Chilisspot Management.....	113

4.2.11 Halaman PostPlan.....	114
4.2.12 Halaman Billing Plan.....	115
4.2.13 Halaman Security	116
BAB V : PENUTUP	117
5.1 Kesimpulan	117
5.2 Saran	117
DAFTAR PUSTAKA	119

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Konsep PHP	19
Gambar 2.2 <i>Format Paket Data Radius</i>	23
Gambar 2.3 Proses User Meminta Layanan Koneksi.....	27
Gambar 2.4 Traffic pesan client NAS Dengan server RADIUS	27
Gambar 2.5 Struktur Standart Chillispot.....	32
Gambar 3.1 Bagan Alir Sistem Proses Login User	37
Gambar 3.2 Bagan Alir Sistem Login Pengelola	38
Gambar 3.3 Bagan Alir Sistem Control Panel PostPlan Administrator	39
Gambar 3.4 Bagan Alir Sistem Control Panel PostPlan Chasier	40
Gambar 3.5 Bagan Alir Sistem Control Panel Chillispot Adminstrator	41
Gambar 3.6 Bagan Alir Sistem Control Panel Chasier Management Administrator.....	42
Gambar 3.7 Bagan Alir Sistem Control Panel Security Management Administrator.....	43
Gambar 3.8 Bagan Alir Sistem Statistic Management Adminstrator.....	44
Gambar 3.9 Bagan Alir Sistem Proses Logout.....	45
Gambar 3.10 Diagram Konteks Billing System RT/RW NET.....	46
Gambar 3.11 Diagram Berjenjang Billing System RT/RW NET	47
Gambar 3.12 DAD Level 1 Proses 1 Billing System RT/RW NET	48
Gambar 3.13 DAD Level 1 Proses 2 Billing System RT/RW NET.....	48
Gambar 3.14 DAD Level 2 Proses 1 Billing System RT/RW NET	49
Gambar 3.15 DAD Level 2 Proses 2 Billing System RT/RW NET.....	50

Gambar 3.16	Desain Struktur Administrator.....	61
Gambar 3.17	Struktur Kasir Desain	63
Gambar 3.18	Desain Login User	65
Gambar 3.19	Desain Login Admin	65
Gambar 3.20	Desain Input Cashier Management.....	66
Gambar 3.21	Desain Input PostPlan Adminstrator	66
Gambar 3.22	Desain Input BillingPlan	67
Gambar 3.23	Desain Input Chillispot.....	67
Gambar 3.24	Desain Input Postpiad Account	68
Gambar 3.25	Desain Input Add Cashier.....	68
Gambar 3.26	Desain Input Change Password Cashier.....	69
Gambar 3.27	Desain Input Voucher Management.....	69
Gambar 3.28	Desain Output Admin Control Panel Home.....	70
Gambar 3.29	Desain Output Manage Cashier.....	70
Gambar 3.30	Desain Output Invoice.....	71
Gambar 3.31	Desain Output Logout User.....	71
Gambar 3.32	Desain Output Statistic.....	72
Gambar 4.1	Jendela Software Source.....	92
Gambar 4.2	Jendela Address Source	93
Gambar 4.3	Jendela Software Source setelah di masukkan address source yang baru.	93
Gambar 4.4	Jendela Synaptic Package Manager.....	95
Gambar 4.5	Jendela Remasterys.....	95
Gambar 4.6	Jendela Remastersys Backup.....	96

Gambar 4.7	jendela Remastersys Backup Confirmation.....	96
Gambar 4.8	Jendela Terminal Program Remastersys Backup	97
Gambar 4.9	Jendela Remastersys Backup selesai	97
Gambar 4.10	Jendela Remastersys Modify	98
Gambar 4.11	jendela Remastersys Clean	98
Gambar 4.12	Halaman add certifacate RT/RW NET Hostpot dengan menggunakan web browser.....	102
Gambar 4.13	Halaman Login	103
Gambar 4.14	Halaman Home Control Panel Cashier.....	104
Gambar 4.15	Halaman Postpaid Account Management.....	105
Gambar4.16	Halaman Voucher Menegement	106
Gambar4.1.7	Halaman Invoice Management.....	107
Gambar 4.18	Halaman Statistik.....	108
Gambar 4.19	Halaman Chasier Change Password	109
Gambar 4.20	Halaman Homr Control Panel Adminstrator	110
Gambar 4.21	Halaman Cashier Management.....	111
Gambar 4.22	Halaman Add Cashier.....	112
Gambar 4.23	Halaman Chillispot Management	113
Gambar 4.24	Halaman PostPlan.....	114
Gambar 4.25	Halaman Billing Plan.....	115
Gambar 4.26	Halaman Security.....	116

DAFTAR TABEL

	Halaman
Tabel 2.1 Simbol-simbol Flowchart	8
Tabel 2.2 Simbol DAD (Data Flow Digram)	19
Tabel 2.3 Tabel Type Paket RADIUS	23
Tabel 3.1 Identifikasi Entity	46
Tabel 3.2 Struktur Tabel Post Plan.....	51
Tabel 3.3 Struktur Tabel User Profile.....	51
Tabel 3.4 Struktur Tabel User	51
Tabel 3.5 Tabel Billing Plan.....	52
Tabel 3.6 Tabel Invoice	53
Tabel 3.7 Tabel invoice Detail.....	55
Tabel 3.8 Tabel Postpaid Account.....	53
Tabel 3.9 Tabel Pospaid Account Bill.....	54
Tabel 3.10 Tabel PostPaid Account List	54
Tabel 3.11 Tabel Postpaid Account List.....	54
Tabel 3.12 Tabel Voucher	55
Tabel 3.13 Tabel Voucher List	55
Tabel 3.14 Tabel Chiispot Sesion.....	56
Tabel 3.15 Tabel NAS (Network Acces Server)	56
Tabel 3.16 Tambel Radius Account	57
Tabel 3.17 Tabel Radius Check.....	58
Tabel 3 18 Tabel Radius Group Check.....	58
Tabel 3.19 Tabel Radius group Replay	58

Tabel 3.20 Tabel Radius Post Authorization.....	59
Tabel 3.21 Tabel radius replay	59

BAB I

PENDAHULUAN

1.1. Latar Belakang

Seiring dengan tuntutan untuk meningkatkan mobilitas dan didukung dengan teknologi radio frekuensi, terciptalah Teknologi WLAN (*Wireless Local Area Network*). *Hotspot* adalah istilah yang digunakan untuk fasilitas *wireless* LAN yang tersedia pada area tertentu. Sementara itu, aplikasi yang banyak digunakan melalui *hotspot* salah satunya adalah Internet. Jadi, dengan *Hotspot*, User bisa berbagi koneksi internet tanpa kabel, *wireless* printing, dan lain-lain.

Mobilitas yang tinggi dari pengguna internet membuat penggunaan *hotspot* menjadi tuntutan. Walaupun secara umum koneksi *wireless* masih belum bisa mengalahkan teknologi pendahulunya (*wired*), perlu dipertimbangkan peningkatan mobilitas yang luar biasa pada teknologi *wireless*. Pertemuan bisnis yang memerlukan koneksi internet dapat dilakukan tidak terbatas di perkantoran saja, tetapi dapat dilakukan di semua *public area* seperti warnet, rumah, atau area *hotspot* yang telah banyak tersedia.

RT/RW Net, demikian istilah yang digunakan oleh pakar Internet Dr. Onno Widodo Poerbo dan Ir. Michael Sunggiardi, Direktur BogorNet (BONET) ISP yang berafiliasi dengan IndoNet. RT/RW Net inilah solusi atau alternatif untuk mengakses Internet bagi rumah tangga dan mengatasi mahal dan langkanya sambungan telepon.

Seiring dengan perkembangan teknologi yang demikian pesatnya maka dibutuhkan sebuah media yang dapat mengelola penggunaan *Hotspot* untuk kepentingan umum khususnya di RT/RW Net yang berbasis web, hal inilah yang mendorong penulis untuk mengambil Skripsi dengan judul **“Pengembangan Aplikasi Billing System Berbasis Open Source di RT/RW Net”**.

1.2 Rumusan Masalah

Dari uraian di atas dapat dirumuskan beberapa masalah yang dihadapi antara lain, sebagai berikut :

1. Diperlukan sebuah fasilitas yang nantinya dapat mengelola penggunaan jaringan *Hotspot*.
2. Diperlukan sebuah manajemen *system* yang berbasis web yang memiliki *interface* yang menarik, serta dapat digunakan dengan mudah dan cepat.

1.3 Batasan Masalah

Batasan masalah dari skripsi ini adalah sebagai berikut :

1. Membahas pembuatan rancangan (*prototype*) *Hotspot Billing System* dan penerapannya yang akan digunakan pada RT / RW NET.
2. Bahasa pemrograman yang digunakan adalah PHP yang dihubungkan dengan MySQL sebagai database, Mamboo, penggunaan web server Apache, RADIUS (*Remote Authentication Dial IN User Service*), Server menggunakan FreeRADIUS Mamboo.
3. Sistem Informasi berbasis Web.
4. Tidak membahas soal jaringan.

1.4 Tujuan

Tujuan dari skripsi ini adalah membuat sebuah Manajemen *Hostpot Billing System* berbasis *Open Source* pada RT/RW NET. Untuk mengelola pemanfaatan internet khususnya jaringan *Hostpot* pada RT / RW NET.

1.5 Metodologi

A. Metode Penelitian

Metode penelitian yang dilakukan adalah sebagai berikut :

- a. Melakukan survey lapangan berupa lokasi yang menjadi titik *Hotspot*.
- b. Melakukan pengamatan dan peninjauan pemanfaatan jaringan *wireless* di suatu kawasan RT/RW.
- c. Mengidentifikasi masalah-masalah yang sering muncul pada penggunaan jaringan *wireless* pada kawasan RT/RW dan bagaimana cara kita untuk memecahkan masalah tersebut.
- d. Melakukan analisa terhadap sistem yang akan dibuat. Hal ini dilakukan agar penulis dapat mengetahui bagaimana prosedur kerja sistem, dan hal-hal apa saja yang dibutuhkan oleh sistem.
- e. Membuat pemodelan sistem berupa bagan alir (*Flow chart*), Diagram Arus Data (*Data Flow-Diagram*), dan ERD (*Entity Relation Diagram*).
- f. Memilih distro linux yang stabil yang dapat digunakan sebagai server pada *Billing System* nantinya.
- g. Melakukan perancangan website yang sesuai dengan kebutuhan. Menggunakan CMS dengan bahasa pemrograman PHP dengan pengolah database MySQL.

B Teknik Pengumpulan Data

Pada penelitian ini penyusun megumpulkan data yang diperlukan dengan menggunakan teknik sebagai berikut :

1. Studi Literatur mengenai konsep Sistem Operasi *Open Source* Ubuntu 8.04 LTS dan juga pengelolaan manajemen jaringan dengan menggunakan FreeRADIUS dan Chillispot.
2. Studi Literatur mengenai konsep *web-server* Apache, dan juga konsep bahasa PHP (*PHP Hypertext Preprocessor*), serta basis data MySQL.
3. Konsultasi dengan pihak pengelola jaringan internet pada salah satu wilayah RT/RW NET.

1.6 Sistem Penulisan

Sistematika penulisan yang diuraikan dalam penyusunan skripsi ini adalah sebagai berikut :

BAB I : PENDAHULUAN

Bab ini berisi tentang latar belakang, tujuan, permasalahan, batasan masalah, dan sistematika pembahasan dari skripsi ini.

BAB II : TINJAUAN PUSTAKA

Bab ini berisi penjelasan tentang teori-teori yang mendukung dalam perancangan sistem billing yang meliputi pengertian dari *Hotspot, Billing, Sistem, Billing Sytem, internet. Open Source*, serta perangkat lunak yang digunakan.

BAB III : PERANCANGAN DAN PEMBUATAN SISTEM

Bab ini berisi tentang perencanaan desain sistem *billing hotspot* dan menjelaskan implementasi yang dilakukan mulai dari perancangan sistem sampai *design user interface*.

BAB IV : HASIL DAN ANALISA

Bab ini berisi pembahasan hasil pengujian dan analisa mengenai cara kerja dari sistem *billing hotspot*.

BAB V : PENUTUP

Bab ini berisi kesimpulan dan saran dari hasil pembahasan pada skripsi ini.

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Billing

Billing berasal dari bahasa Inggris yaitu *bill (noun)*, yang artinya bukti transaksi pembayaran. Maka *billing (adv)* dapat juga diartikan mengirimkan bukti transaksi, atau mengumumkan bukti transaksi.

2.2 Pengertian Sistem (*System*)

Sistem berasal dari bahas latin (*systema*) dan bahasa Yunani (*sustema*) adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Istilah ini sering dipergunakan untuk menggambarkan suatu set entitas yang berinterksi, dimana suatau model matematika seringkali bias dibuat.

Sistem juga merupakan keasatuan bagian-bagian yang saling berhubungan yang berada dalam suatu wilayah serta memiliki item-item penggerak (*manetsch* dan *Park*, 1979 dalam Eriyanto, 1999).

Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan (McLeod, 1995 : 13).

Suatu sistem adalah suatu jaringan kerja dari prosedur - prosedur yang saling berhubungan, berkumpul bersama – sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. (Jogiyanto,1999 : 1).

2.2.1 Pemodelan Sistem

Model adalah bentuk yang mewakili suatu realitas. Bentuk untuk mewakili suatu realitas tersebut dapat berbentuk prosedur yang disebut dengan model prosedur, dapat berbentuk logika yang disebut dengan model logika.












Pemodelan akan sangat menentukan *output* yang dihasilkan oleh sistem informasi. Pemilihan dan pembentukan model yang tidak tepat akan menghasilkan *output* yang juga tidak akan mengena. Oleh sebab itu pemodelan harus betul-betul dirancang dengan baik dan benar.


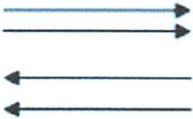
Alasan perlunya pemodelan sistem yaitu dapat memfokuskan perhatian pada hal-hal penting dalam sistem tanpa terlibat terlalu jauh, mendiskusikan perubahan dan koreksi terhadap kebutuhan pemakai dan membantu pendesain sistem dalam membuat suatu program.

Model-model Sistem :

- a. Bagan alir (*Flowchart*) adalah bagan yang menunjukkan alir di dalam program atau prosedur sistem secara logika. *Flowchart Program* dibuat dan derivikasi *Flowchart System*. Beberapa simbol dalam *Flowchart System* dapat juga digunakan untuk bagan alir program, diantaranya yaitu simbol penghubung, simbol titik terminal, simbol proses dan simbol garis alir. Simbol-simbol yang digunakan dapat di gambarkan sebagi berikut :





Table 2.1 Simbol-simbol *Flowchart*^[3]

Nama Simbol	Simbol	Keterangan
Dokumen		Menunjukkan dokumen input dan output baik untuk proses manual, mekanik atau komputer
Terminal (Mulai/berhenti)		Menunjukkan awal kegiatan dan akhir kegiatan suatu program
Input / Output		Untuk mewakili data <i>input</i> dan menuliskan outputnya
Proses		Simbol yang melambangkan diprosesnya suatu data
Predefined		Menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain
Clerical Option (Manual Check)		Digunakan untuk menjelaskan adanya proses manual yang terjadi proses tersebut
Diskette (Disket)		Menunjukkan input / output menggunakan disket
Explanation (Penjelasan)		Menunjukkan penjelasan dari suatu proses
Direct Access Device		Dipakai untuk menandakan adanya media <i>direct acces</i> (hardisk) yang digunakan untuk pengoperasikan file pada tahap tersebut
Display		Simbol digunakannya layer monitor untuk menyajikan suatu informasi tertentu.
Online Keyboard		Menyatakan adanya interversi proses yang harus diberikan oleh operator, pada saat berlangsungnya suatu rangkaian proses di computer.

<p>Connector (Penghubung)</p>		<p>Bila suatu flow-chart sangat panjang dan diputus ditengah sebelum selesai, jika disambung dalam halaman yang sama maka digunakan simbol ini(1). Bila disambung pada halaman yang lain digunakan simbol ini(2).</p>
<p>Flow – Lines (Garis alir)</p>		<p>Menunjukkan arah selanjutnya yang akan dituju. Garis4garis ini juga yang menghubungkan simbol4simbol dari Flow-Chart.</p>

- b. HIPO (*Hierarchy Plus Input-Proses-Output*) adalah alat desain dan teknik dokumentasi program dalam siklus pengembangan sistem.
- c. Tabel keputusan (*Decision Table*) adalah tabel yang digunakan sebagai alat bantu untuk menyelesaikan logika dalam program.
- d. Diagram Arus Data (*Data Flow-Diagram*), menggambarkan suatu sistem yang telah ada atau sistem baru akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan. Beberapa simbol yang digunakan dalam DAD dapat digambarkan sebagai berikut :

Tabel 2.2 Simbol DAD (*Data Flow Diagram*)^[3]

Simbol	Keterangan
	Kesatuan luar (<i>External Entity</i>) merupakan kesatuan dilingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan <i>input</i> atau menerima <i>output</i> dari system
	Arus data (<i>Data Flow</i>) merupakan masukan untuk sistem hasil dari proses sistem.
	Proses (<i>Process</i>) merupakan kegiatan atau kerja yang dilakukan oleh orang, mesin atau computer dari hasil suatu arus data yang masuk kedalam proses. Suatu proses harus menerima arus data dan menghasilkan arus data.
	Simpanan data (<i>Data Storage</i>), berupa <i>file / database</i> di sistem komputer, arsip atau catatan manual. Namun data <i>Storage</i> menunjukkan nama dari <i>filenya</i> .

- e. ERD (*Entity Relation Diagram*) adalah menggambarkan hubungan antara obyek data sekaligus notasi yang digunakan untuk melakukan bukan aktifitas pemodelan data.
- f. Kamus data (KD) atau *data dictionary* atau *system dictionary* adalah katalog fakta tentang data dan kebutuhan informasi dari suatu system dapat mendefinisikan data yang mengalir dalam sistem dengan lengkap. KD dibuat berdasarkan arus data yang ada di DAD/DFD. KD harus dapat mencerminkan keterangan yang jelas tentang data yang dicatatnya karena KD sangat diperlukana dalam perancangan *database* nanti

2.2.2 Rancangan Basis Data (*Database*)

Database adalah kumpulan *file-file* yang saling berelasi, relasi tersebut biasa ditunjukkan dengan kunci dari tiap *file* yang ada. Satu *database* menunjukkan satu kumpulan data. *Database* berfungsi sebagai basis penyedia informasi bagi pemakainya. Tahap desain secara umum sebelumnya, desain *database* hanya dimaksudkan untuk mengidentifikasi kebutuhan *file-file database* yang diperlukan oleh system informasi saja. Pada tahap desain terinci ini desain *database* dimaksudkan untuk mendefinisi isi dan struktur dari tiap-tiap yang telah diidentifikasi didesain secara umum. Elemen-elemen data dalam suatu *file database* harus dapat digunakan untuk pembuatan suatu *output*. Isi atau struktur dari suatu *file database* tergantung dari arus data masuk dan keluar ke atau dari *file* tersebut. Arus data yang mengalir dapat dilihat pada DAD/DFD.

2.2.3 Rancangan *Input*

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

2.2.4 Rancangan *Output*.

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supersistem. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

2.3 Pengertian *Billing System* (Sistem *Billing*)

Sistem *Billing* merupakan system yang membantu para usahawan untuk mengatur dan mencatat segala transaksi yang terjadi. Contoh bagi pengusaha warung internet, *billing system* digunakan untuk memonitor pengguna dan pemasukan warnetnya. Sedangkan jika bagi usahawan dibidang perumhaskitan system *billing* digunakan untuk mencatat proses pelayanan, mulai pasien datang sampai dengan pasien pulang. Menghitung biaya yang harus dibayar pasien secara otomatis, serta memberikan informasi sebagai analisa pengambilan keputusan secara cepat dan akurat.

2.4 Pengertian Internet

Internet adalah sistem global yang saling berhubungan jaringan komputer yang menggunakan standar *Internet Protocol Suite* (TCP / IP) untuk melayani jutaan pengguna di seluruh dunia. Ini adalah jaringan dari jaringan yang terdiri dari jutaan pribadi dan umum, akademik, bisnis, dan jaringan pemerintah lokal lingkup global yang dihubungkan oleh sebuah *array* luas elektronik dan teknologi jaringan

optik. Internet membawa *array* yang luas dari sumber informasi dan pelayanan, terutama terkait antar dokumen *hypertext* dari *World Wide Web* (WWW) dan infrastruktur untuk mendukung surat elektronik

Kebanyakan media komunikasi tradisional, seperti telepon dan layanan televisi, yang mengubah bentuk atau mengulang dengan menggunakan teknologi Internet, sehingga menimbulkan layanan seperti *Voice over Internet Protocol* (VoIP) dan IPTV. Penerbitan surat kabar telah dibentuk kembali ke situs *Web*, *blog*, dan *web feed*. Internet telah diaktifkan atau mempercepat penciptaan bentuk-bentuk baru interaksi manusia melalui *instant messaging*, *Internet forum*, dan situs jaringan sosial.

Asal-usul Internet mencapai kembali ke tahun 1960-an ketika Amerika Serikat dibiayai proyek riset dari badan-badan untuk membangun militer yang kuat, kesalahan-toleran dan didistribusikan jaringan komputer. Penelitian ini dan periode pendanaan sipil AS yang baru tulang punggung oleh *National Science Foundation* melahirkan partisipasi di seluruh dunia dalam pengembangan teknologi jaringan baru dan mengarah ke komersialisasi jaringan internasional pada pertengahan 1990-an, dan mengakibatkan popularisasi berikut yang tak terhitung jumlahnya aplikasi dalam hampir setiap aspek kehidupan manusia modern. Seperti tahun 2009, diperkirakan seperempat dari penduduk Bumi menggunakan layanan dari Internet.

Internet tidak memiliki pemerintahan yang terpusat baik dalam implementasi teknologi atau kebijakan untuk akses dan penggunaan; masing-masing jaringan *konstituen* standar menetapkan sendiri. Hanya definisi berlebihan dari dua ruang nama utama di Internet, *Internet Protocol* ruang dan *Domain Name System*, yang

diarahkan oleh organisasi pengelola, *Internet Corporation for Assigned Names and Numbers* (ICANN). Fondasi teknis dan standardisasi protokol inti (IPv4 dan IPv6) adalah suatu kegiatan *Internet Engineering Task Force* (IETF), sebuah organisasi nirlaba yang berafiliasi longgar-peserta internasional bahwa siapa pun dapat mengaitkan dengan ikut serta keahlian teknis.

2.4.1 *World Wide Web*

World Wide Web adalah suatu sarana pembagian informasi antar pengguna jaringan computer. Dengan adanya *World Wide Web* maka informasi (berupa tulisan, gambar, suara, video) dapat diberikan kepada orang lain sehingga memungkinkan terjadinya pertukaran pengetahuan atau pembagian pekerjaan.

World Wide Web (biasa disingkat *WWW*) adalah salah satu dari sekian banyak layanan yang ada di internet. Layanan ini paling banyak digunakan di internet untuk menyampaikan informasi karena sifatnya mendukung *multimedia*. Artinya informasi tidak hanya bias disampaikan melalui teks, tapi juga dengan gambar, suara dan video. *Web-Server* merupakan *server* internet yang mampu melayani koneksi transfer data dalam protocol HTTP.

Web-server merupakan hal terpenting sari *server* di internet dibandingkan server lainnya, seperti *e-mail server*, *FTP server*, ataupun *news server*. Hal ini di sebabkan *webserver* telah dirancang untuk dapat melayani beragam jenis data, dari teks hingga grafis tiga dimensi. Kemampuan ini telah menyebabkan serbagai institusi seperti perusahaan ataupun universitas dapat menerima kehadirannya dan juga sekaligus menggunakannya sebagai sarana di internet.

Web-Client adalah komputer yang tergantung dalam jaringan atau internet yang meminta informasi. Untuk dapat mengakses *web-server*, *web-client* menggunakan suatu aplikasi khusus yang dinamakan dengan *web-browser*. *Web-Browser* meminta dan menerima data dari *web-server* melalui suatu protocol yang di sebut HTTP (*Hypertext Trasfer Protokol*). Protokol ini bertugas mengirimkan perintah dari *web-browser* ke *web-server* serta mengirimkan data yang diminta oleh *web-browser*. *File* yang dikirim dalam layanan web ini dapat berupa **.asp,*.htm,*.html,*.php*, dan lain sebagainya.

2.5 Pengertian *Open Source*

Open Source dalam bahasa Indonesia adalah *source* yang terbuka. *Open Source Software* atau lebih dikenal hanya *Open Source* adalah istilah yang digunakan untuk *software* yang membuka/ membebaskan *source codenya* untuk dilihat oleh orang lain dan membiarkan orang lain mengetahui cara kerja *software* tersebut dan sekaligus memperbaiki kelemahan-kelemahan yang ada pada *software* tersebut.

Menurut sistus resmi GNU (GNU's Not Unix) *Open Souce* mengacu pada kebebasan para penggunanya untuk menjalankan, menggandakan, menyebarkan, mempelajari, mengubah dan meningkatkan kinerja perangkat lunak. Tepatnya mengacu pada empat jenis kebebasan bagi para pengguna perangkat lunak :

1. Kebebasan untuk menjalankan programnya untuk tujuan apa saja (kebebasan 0).

2. Kebebasan untuk mempelajari bagaimana program itu bekerja serta dapat disesuaikan dengan kebutuhan anda (kebebasan 1). Akses pada kode program merupakan suatu prasyarat.
3. Kebebasan untuk menyebarluaskan kembali salinan perangkat lunak tersebut sehingga dapat membantu sesama anda (kebebasan 2).
4. Kebebasan untuk meningkatkan kinerja program dan dapat menyebarkannya ke khalayak umum sehingga semua menikmati keuntungan (kebebasan 3). Akses pada kode program merupakan suatu prasyarat juga.

Open Source adalah sistem pengembangan yang tidak di koordinasi oleh suatu orang/ lembaga pusat, tetapi oleh para pelaku yang berkerja sama dengan memanfaatkan kode sumber (*source-code*) yang tersebar dan tersedia bebas (biasanya menggunakan fasilitas komunikasi internet). Pola pengembangan ini mengambil model ala *bazaar*, sehingga pola *Open Source* ini memiliki ciri bagi komunitasnya yaitu adanya dorongan yang bersumber dari budaya memberi, yang artinya ketika suatu komunitas menggunakan sebuah program (*Open Source*) dan telah menerima sebuah manfaat kemudian akan termotivasi untuk menimbulkan sebuah pertanyaan apa yang bisa pengguna berikan balik kepada orang banyak.

Pengembangan Linux bersifat open source artinya source code dari aplikasi pembentuk sistem dan aplikasi lainnya diberikan secara terbuka sehingga setiap orang dapat melakukan modifikasi dan kustomisasi sesuai dengan kebutuhan masing-masing. Pola *Open Source* lahir karena kebebasan berkarya, tanpa intervensi berpikir dan mengungkapkan apa yang diinginkan dengan menggunakan

pengetahuan dan produk yang cocok. Kebebasan menjadi pertimbangan utama ketika dilepas ke publik. Komunitas yang lain mendapatkan kebebasan untuk belajar, mengutak-atik, merevisi ulang, membenarkan ataupun bahkan menyalahkan tetapi kebebasan ini juga datang bersama dengan tanggung jawab, bukan bebas tanpa tanggung jawab. Pada intinya konsep sumber terbuka adalah membuka "kode sumber" dari sebuah perangkat lunak. Konsep ini terasa aneh pada awalnya dikarenakan kode sumber merupakan kunci dari sebuah perangkat lunak. Dengan diketahui logika yang ada di kode sumber, maka orang lain semestinya dapat membuat perangkat lunak yang sama fungsinya. Sumber terbuka hanya sebatas itu. Artinya, dia tidak harus gratis. Definisi sumber terbuka yang asli adalah seperti tertuang dalam OSD (*Open Source Definition*)/ *Definisi sumber terbuka*. Pergerakan perangkat lunak bebas dan sumber terbuka saat ini membagi pergerakannya dengan pandangan dan tujuan yang berbeda. Sumber terbuka adalah pengembangan secara metodologi, perangkat lunak tidak bebas adalah solusi suboptimal. Bagi pergerakan perangkat lunak bebas, perangkat lunak tidak bebas adalah masalah sosial dan perangkat lunak bebas adalah solusi.

2.6 Perangkat Lunak yang digunakan

2.6.1 Apache

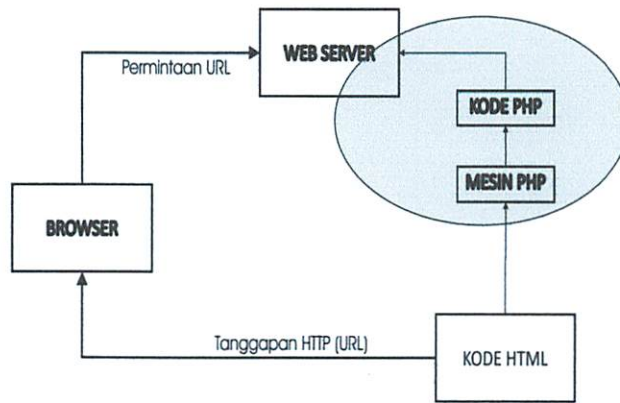
Salah satu perangkat lunak yang biasa digunakan untuk *web-server*. *Apache* merupakan perangkat lunak *open source* yang sekarang ini sudah merebut pasar dunia lebih dari 50 %. *Web-server* ini fleksibel terhadap berbagai system operasi, dalam penggunaannya *Apache* merupakan turunan dari *web-server* yang dikeluarkan oleh NCSA HTTPd 1.3 pada sekitar 1995. kelebihan *web-server Apache* antara lain:

- a. *Freeware* (perangkat lunak yang gratis).
- b. Mudah di *install*.
- c. Mampu beroperasi di berbagai *platform* sistem operasi.
- d. Mudah mengonfigurasinya.
- e. Mampu di kompilasi sesuai dengan spesifikasi HTTP yang ada sekarang ini.
- f. Menyediakan sebuah fitur bernama *virtual-server*.
- g. Lebih aman karena menggunakan level-level pengamanan dan juga teknik enkripsi.
- h. Komplikasi yang tinggi terhadap berbagai *platform*.
- i. Mendukung *third-party* berupa modul-modul tambahan.
- j. Mempunyai *technical support* melalui web.

2.6.2 PHP (*PHP Hypertext Preprocessor*)

PHP adalah salah satu bahasa *server-side* yang didesain khusus untuk aplikasi web. PHP dapat disisipkan di antara bahasa HTML dan karena bahasa *server-side*, maka bahasa PHP akan dieksekusi di server, sehingga yang dikirimkan ke web-browser adalah "hasil jadi" dalam bentuk HTML, dan kode PHP tidak akan terlihat.

Secara khusus, PHP dirancang untuk membentuk web dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya anda bisa menampilkan isi *database* kehalaman *web*. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti *ASP*, *ColdFusion* ataupun *Perl*.



Gambar 2.1 Konsep PHP ^[2]

PHP dibuat pertama kali oleh Rasmus Lerdorf, yang pada awalnya dibuat untuk menghitung jumlah pengunjung pada homepagenya pada tahun 1994. Skrip ini selanjutnya dikemas menjadi *tool* yang disebut "*Personal Home Page*". Paket inilah yang menjadi cikal bakal PHP. Pada tahun 1995, Ramus menciptakan *PHP/FI* versi 2. Pada versi inilah pemrograman dapat menempelkan kode terstruktur ke dalam tag *HTML*. Yang menarik, kode PHP juga bisa berkomunikasi dengan *database* dan melakukan perhitungan-perhitungan yang kompleks sambil jalan.

Hal terbaik dari PHP adalah kemudahan untuk dipelajari terutama bagi *programmer* pemula, namun memiliki fitur yang bisa dikembangkan secara professional. PHP juga mampu mengakses *database*, namun sangat direkomendasikan menggunakan *database* MySQL dan menggunakan *server Apache*.

PHP dapat berjalan di berbagai sistem operasi (*multi platform*), keunggulan lainnya adalah PHP juga mendukung komunikasi dengan layanan seperti protocol IMAP,SMTP,POP3. PHP dapat di-*instal stand-alone* ataupun sebagai bagian dari

Apache web-server. Selain itu, PHP termasuk dalam *Open Souce Product*, yang berarti *source codenya* dapat dimodifikasi dan di distribusikan secara bebas. PHP juga di edarkan secara gratis (*freeware*.)

2.6.3 My SQL.

SQL (*Structured Query Language*) adalah bahasa standar yang digunakan untuk mengakses *server* basis data. Sejak tahun 804an bahasa ini telah dikembangkan oleh IBM, dan kemudian diikuti dengan adanya *Oracle*, *Informix*, dan *Sybase*. Dengan menggunakan SQL, proses akses basis data menjadi lebih *user-friendly* dibandingkan dengan *dBase* ataupun *Clipper* yang masih menggunakan perintah-perintah pemrograman murni.

MySQL sendiri merupakan *database server* dari SQL yang bersifat *multi-user* dan *multi-threaded*. MySQL dibuat oleh TcX dan telah dipercaya mengelola sistem dengan 40 buah basis data, yang berisi 10 ribu tabel dan 500 diantaranya 7 juta baris (kurang lebih 100 *Gigabyte* data). Basis data ini dibuat untuk keperluan sistem basis data yang cepat, handal, efisien, dan mudah digunakan. Beberapa keunggulan dari MySQL antara lain sebagai berikut :

- a. *Freeware*, yang berarti aplikasi ini bebas untuk di *download* dan disebarluaskan, meskipun ada beberapa diantaranya yang bersifat *shareware* untuk lingkungan Windows, salah satunya adalah *SQL-Server*.
- b. Dapat berjalan hampir di semua *platform*, baik itu di Windows, Unix (Linux), ataupun Mac.

- c. Merupakan program yang *multi-user*, artinya bisa digunakan lebih dari satu orang dengan hak akses (*privileges*) yang berbeda-beda.
- d. Bersifat *multi-threaded*, sehingga dapat dipasang pada *server* yang memiliki *multi-CPU*.
- e. Mendukung berbagai bahasa pemrograman, seperti Visual Basic, C, C++, Java, Perl, Python, dan lain4lain.
- f. Memiliki sistem keamanan yang cukup baik dengan metode verifikasi *host*.
- g. Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi sistem basis data.
- h. Mendukung *record* yang memiliki kolom dengan panjang tetap atau bervariasi.

Sistem basis data di MySQL sendiri memiliki sistem sekuritas dengan verifikasi, yaitu *username*, *password*, dan *host*. Verifikasi *host* memungkinkan untuk membuka sekuriti di *localhost*, tetapi tertutup bagi *host* lain. Sistem sekuriti ini ada di dalam basis data MySQL dan pada tabel pengguna. Proteksi juga bisa dilakukan terhadap basis data, tabel, hingga kolom secara terpisah.

2.6.4 FreeRadius

Perubahan yang terjadi pada suatu jaringan tidak dapat diabaikan, bila sebuah modem ditambahkan pada sebuah komputer atau server komunikasi pada jaringan perusahaan untuk melakukan pengendalian jaringan atau pengaksesan internet menggunakan jaringan perusahaan, hal ini menyebabkan

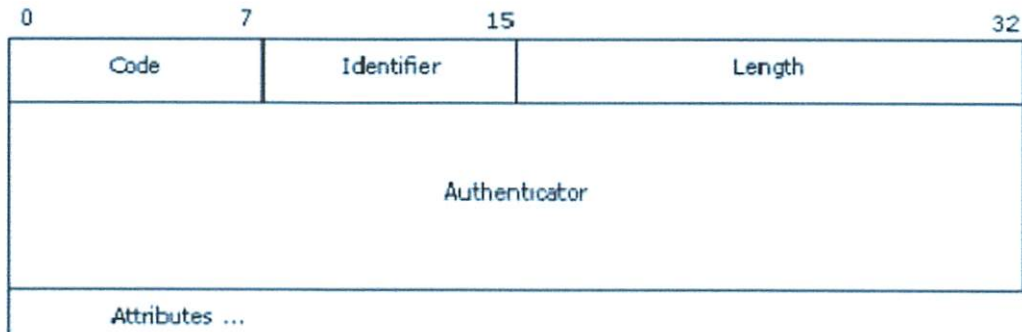
jaringan akan lebih rentan terhadap adanya pelanggaran keamanan. Maka untuk menjamin keamanan suatu sistem dari penyalahgunaan oleh orang yang tidak bertanggung jawab dibutuhkan adanya sistem autentikasi yang baik. Untuk menyimpan data pengguna yang mempunyai hak yang jumlahnya sangat banyak dibutuhkan suatu media penyimpanan yang besar. Untuk sistem *embeded* yang terbatas media penyimpanan yang dimiliki seperti *router*, *server*, *modem*, *switch* dan lain-lain, maka diperlukan sistem autentikasi yang tidak pada sistem itu sendiri, dengan kata lain diperlukan sistem autentikasi diluar sistem *embeded* tersebut, sistem ini dinamakan *Remote Access Dialin User Service (RADIUS)*.

RADIUS atau *Remote Authentication Dial-In User Service* merupakan sebuah protocol yang memungkinkan perusahaan untuk melakukan *Authentication* (pembuktian keaslian), *Authorize* (otoritas/pemberian hak) dan *Accounting* (akutansi) (AAA) untuk *me-remote* para pengguna atau *user* yang ingin mengakses suatu sistem atau layanan dari pusat *server* jaringan komputer (C. Rigney, S. Willens, A. Rubens and W. Simpson, 2001).

Sedangkan FreeRADIUS merupakan protokol server RADIUS yang termasuk dalam OSS (*Open Source Software*). Pada awalnya perkembangan RADIUS menggunakan port 1645 yang ternyata bentrok dengan layanan *datametrics*. Sekarang port yang dipakai RADIUS adalah port 1812 yang format standarnya ditetapkan pada *Request for Command (RFC) 2138*.

a. Format Paket Data RADIUS

Format paket RADIUS terdiri dari *Code*, *Identifier*, *Length*, *Authenticator* dan *Attributes*.



Gambar 2.2 Format Paket data Radius^[3]

- **Code**

Code memiliki panjang 1 byte (8bit), digunakan untuk membedakan *type* pesan RADIUS yang dikirim, tipe pesan RADIUS dapat berupa *access request*, *access accept*, *access reject* dan *access challenge*.

Tabel 2.3 Tabel tpe Paket RADIUS^[3]

Value	Description
1	Access-Request
2	Access-Accept
3	Access-Reject
4	Accounting-Request
5	Accounting-Respond
11	Access-Challenge
12	Status-Server (experimental)
13	Status-Client (experimental)
255	Reserved

Jika paket yang diterima berisi code yang tidak sesuai dengan nilai-nilai di atas, maka paket tersebut akan dibuang.

Access-Request Merupakan paket yang dikirim pada *server* RADIUS dari RADIUS *client*, yang berisi informasi yang digunakan untuk menentukan apakah *user* berhak menggunakan layanan pada NAS yang diakses (seperti atribut *user name*, *userpassword/chap password/pappassword*, *NASIPAddress*, *NASIdentifier*, *NASport*, *NASport type*) dan jenis layanan yang membatasi atau membebaskan *user* tersebut.

Access-Accept Merupakan paket yang dikirimkan oleh *server* RADIUS pada RADIUS *client*, yang berisi informasi konfigurasi spesifik yang diperlukan untuk memulai layanan pada *user*. Dalam mengirim *access-accept*, bagian *identifier* dan *response authenticator* harus berisi tanggapan yang sesuai dengan *access-request* yang tertunda sebelumnya, jika paket tidak sesuai akan dibuang.

Access-Reject Merupakan paket yang dikirim oleh *server* RADIUS pada RADIUS *client*, jika terdapat nilai yang tidak sesuai dengan data yang ada pada *server* RADIUS dan dapat berisi satu atau lebih *Reply Message* dengan informasi berupa teks yang memberitahukan pada *user*, bahwa *user* tersebut tidak dapat terkoneksi pada NAS.

Access-Challenge Server RADIUS dapat di konfigurasi agar memberikan *AccessChallenge* dari *AccessRequest* yang dikirimkan dari NAS, jika NAS tidak mendukung *challenge/response* maka akan diterima sebagai *accessreject* dari server RADIUS, sedangkan jika mendukung *challenge/response* maka *accessrequest* akan dikirimkan kembali, NAS akan menampilkan informasi berupa teks yang meminta tanggapan dari *user*, yang kemudian akan mengirimkan *accessrequest* dengan *identifier* dan *request authenticator* yang baru, dengan *userpassword* yang digantikan oleh tanggapan *user* yang telah dienkripsi.

- **Identifier**

Memiliki panjang 1 *byte* yang digunakan untuk menyesuaikan antara paket permintaan dan respon dari server RADIUS. Server RADIUS dapat mendeteksi jika terdapat permintaan yang sama dari pengguna dengan sumber alamat IP dan port UDP yang sama.

- **Length**

Memiliki panjang 2 *byte*, memberikan informasi mengenai panjang paket. Jika paket kurang atau lebih dari yang diidentifikasi pada *length* maka paket akan dibuang.

- **Authenticator**

Memiliki panjang 16 *byte*, *most significant bit* (MSB) yang pertama kali dikirimkan. Nilai ini digunakan untuk mengautentikasi tanggapan dari

server RADIUS, dan juga digunakan dalam algoritma *password hiding*. Jika pada *access request* dinamakan *request authenticator* yang berisi 128 bit bilangan acak yang tidak dapat diprediksi dan unik dihasilkan oleh *one way hash* MD5 dari sandi rahasia antara *Network Access Server* (NAS) dengan *server* RADIUS. Sedangkan pada *accessaccept*, *accessreject*, dan *accesschallenge* dinamakan *response authenticator* yang berisi *one way hash* MD5 yang dihasilkan dari perhitungan serangkaian bit-bit yang terdiri atas paket RADIUS, *code*, *identifier*, *length*, *request authenticator* dari paket *access request*, dan atribut respon yang diikuti oleh sandi rahasia.

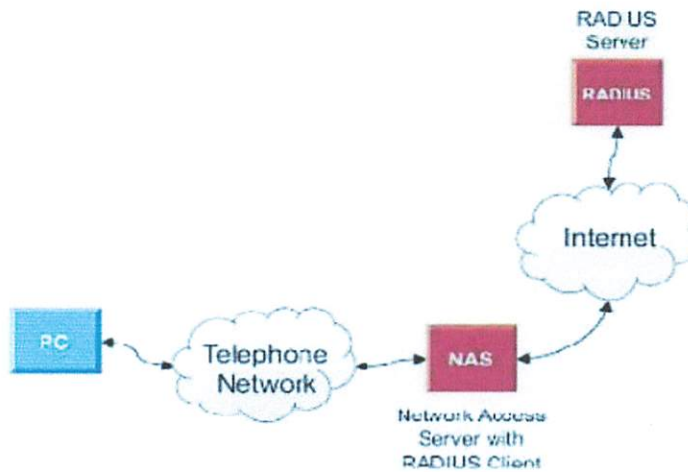
- ***Attributes***

Memiliki panjang yang tidak tetap, berisi autentikasi, otorisasi dan informasi. Contoh atribut RADIUS yaitu, *username* dan *password*.

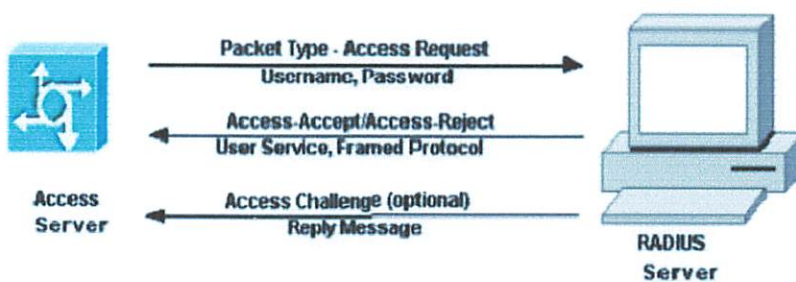
b. Prinsip Kerja RADIUS

RADIUS merupakan protokol *security* yang bekerja menggunakan sistem *clientserver* terdistribusi yang banyak digunakan bersama AAA untuk mengamankan jaringan pengguna yang tidak berhak.

Lalu lintas pesan pada RADIUS menggunakan metode permintaan dan tanggapan (*client/server*).



Gambar 2.3 Proses User Meminta Layanan Koneksi^[5]



Gambar 2.4 Traffic pesan client NAS dengan server RADIUS^[5]

- *User* melakukan dial-in pada NAS. NAS akan meminta *user* memasukkan nama dan *password* jika koneksi tersebut berhasil dibangun.
- NAS akan membangun paket data berupa informasi, yang dinamakan *access request*. Informasi diberikan oleh NAS pada *server* RADIUS berisi informasi spesifik dari NAS itu sendiri yang meminta *access request*, port yang digunakan untuk koneksi modem serta nama dan *password*. Untuk proteksi dari *hackers*, NAS yang bertindak sebagai RADIUS *client*, melakukan enkripsi *password* sebelum dikirimkan pada

RADIUS *server*. *Access request* ini dikirimkan pada jaringan dari RADIUS *client* ke RADIUS *server*. Jika RADIUS *server* tidak dapat dijangkau, RADIUS *client* dapat melakukan pemindahan *route* pada *server* alternatif jika didefinisikan pada konfigurasi NAS.

- Ketika *access request* diterima, *server* autentikasi akan memvalidasi permintaan tersebut dan melakukan dekripsi paket data untuk memperoleh informasi nama dan *password*. Jika nama dan *password* sesuai dengan basis data pada *server*, *server* akan mengirimkan *access accept* yang berisi informasi kebutuhan sistem *network* yang harus disediakan oleh *user*. Selain itu *access accept* ini dapat berisi informasi untuk membatasi akses *user* pada jaringan. Jika proses login tidak menemui kesesuaian, maka RADIUS *server* akan mengirimkan *access reject* pada NAS dan *user* tidak dapat mengakses jaringan.
- Untuk menjamin permintaan *user* benar-benar diberikan pada pihak yang benar, RADIUS *server* mengirimkan *authentication key* atau *signature* yang menandakan keberadaannya pada RADIUS *client*.

RADIUS melakukan autentikasi user melalui serangkaian komunikasi antara *client* dan *server*. Bila *user* telah berhasil melakukan autentikasi, maka *user* tersebut dapat menggunakan layanan yang disediakan oleh jaringan.

Protokol RADIUS merupakan protokol *connectionless* berbasis UDP yang tidak menggunakan koneksi langsung. Satu paket RADIUS ditandai dengan *field* UDP yang menggunakan port 1812. Beberapa pertimbangan

RADIUS menggunakan lapisan transport UDP yaitu:

1. Jika permintaan autentikasi pertama gagal, maka permintaan kedua harus dipertimbangkan.
2. Bersifat *stateless* yang menyederhanakan protokol pada penggunaan UDP.
3. UDP menyederhanakan implementasi dari sisi *server*. Beberapa kelebihan yang diberikan oleh protokol RADIUS yaitu :
 - Menjalankan sistem administrasi terpusat.
 - Protokol *connectionless* berbasis UDP yang tidak menggunakan koneksi langsung.
 - Mendukung autentikasi *Password Authentication*
 - Protocol (PAP) dan *Challenge Handshake*
 - *Authentication Protocol (CHAP) Password* melalui PPP.

Pada protokol RADIUS masih ditemukan beberapa kelemahan seperti :

- Tidak adanya autentikasi dan verifikasi terhadap *access request*.
- Tidak sesuai digunakan pada jaringan dengan skala yang besar.

c. Otentifikasi dan Enkripsi

Otentifikasi dapat dipakai pada beberapa tingkatan dengan menggunakan kombinasi dari beberapa metoda. Sebagai contoh dengan menggunakan EAP4TLS yang otentifikasinya berdasarkan pada standar keamanan 802.1x. EAP4TLS adalah IETF standar untuk metode autentikasi

(RFC2716) yang didukung oleh semua vendor.

Menggunakan *protocol* TLS (*Transort Layer Security*) (RFC 2246) dimana standar paling baru dari *protocol* SSL (*secure Socket Layer*), digunakan untuk keamanan laulintas data pada web dan dibuat pertama kali oleh *netscape*. EAP4TLS menggunakan *Remote Authenticartion Dial4 in User Service* (RADIUS) untuk mengontrol *user* mengakses jaringan nirkabel. Selain itu solusi EAP4TLS ini menggunakan sertifikat digital untuk otentifikasi dari sisi RADIUS *server* maupun *client*. EAP4TLS dibuat berdasarkan pada X.509 *certificates* untuk menangani autentikasi dan membutuhkan PKI (*Public Key Infrastructure*). *Supplicant* harus mempunyai sertifikat yang akan divalidasi oleh *authentication server*.

EAP4TLS menyediakan *mutual authentication* yang kuat antara *supplicant* dan *authentication server* (hal ini hanya benar jika kedua bagian data memvalidasi sertifikat lainnya. *EAP4TLS* *generate dynamic WEP* (*shared secret*) setelah proses pertukaran, sehingga *supplicant* dan *authenticator* dapat melakukan komunikasi yang aman berdasarkan *perpacket authenticated*. Untuk itu dibutuhkan pembuatan PKI (*Public Key Infrastructure*) untuk membuat sertifikat tersebut di atas. Aplikasi untuk membuat sertifikat ini diantaranya openssl.

Openssl adalah *software open source* untuk mengimplementasikan *protokol Secure Socket Layer* (SSL) dan *Transport Layer Security* (TLS) dan sebagai *Certification Authority* (CA) bagi *server* dan *client*.

Proses autentikasi EAP4TLS berlangsung setelah *supplicant*

mengirim pesan *EAPResponse Identity* ke *access point*, dengan *EAPrequest*, *authentication server* mengirim sertifikat ke *supplicant* dan meminta sertifikat dari *supplicant*. Setelah itu *supplicant* memvalidasi sertifikat *server* dan sebagai bagian dari respon EAP, mempersiapkan sertifikat dan juga memulai negosiasi untuk spesifikasi kriptografi. *RADIUS Server* memvalidasi sertifikat *client* dan merespon dengan spesifikasi kriptografi untuk *session*.

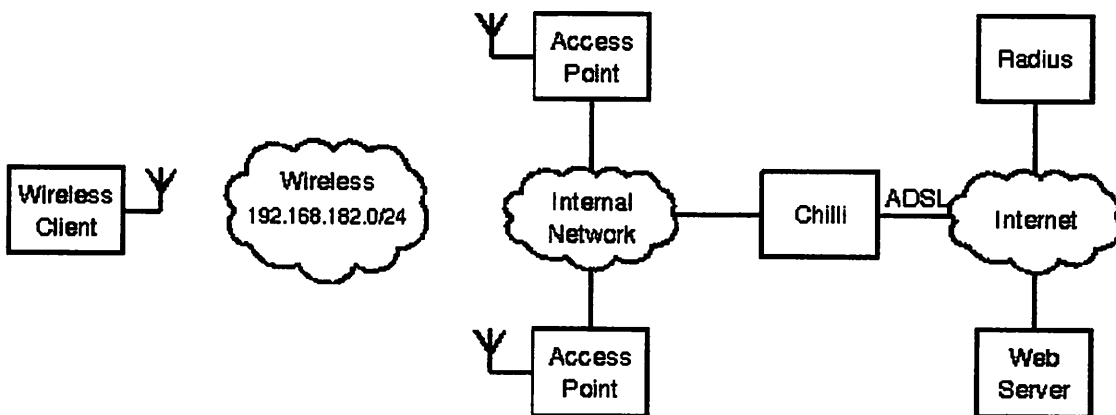
Proses selanjutnya *TLS handshake* antara *authentication server* dan *client*, adalah *generate premaster secret*, mengenkripsinya dengan *server public key* dan mengirim *premaster secret* ke *server* untuk *generate master secret* yang digunakan untuk membuat *secure chanel*. Oleh sebab itu meskipun *TLS* telah benar-benar *setup chanel* terenkripsi antara *authentication server* dengan *supplicant*, *chanel* ini tidak digunakan (*supplicant* ingin berkomunikasi dengan *authenticator*, tidak dengan *authentication server*). *Instead* sebuah *key* dibuat selama proses *session TLS* untuk *chanel* tersebut yang dikirim kepada *authenticator*. Kemudian *supplicant* yang telah mengetahui *TLS secret key*) dan *authenticator* menggunakan *key* tersebut untuk mengamankan komunikasi dengan enkripsi WEP.

2.6.5 Chillispot

Chillispot merupakan *software* untuk implementasi *captive portal*. *Chillispot* biasanya digunakan sebagai otentikasi pada pengguna *wireless LAN*, namun juga bisa digunakan pada jaringan kabel. *Chillispot* mendukung akses masuk berbasis web yang biasa digunakan oleh publik dan *chillispot* juga mendukung WPA atau *WiFi Protected Access*, yang dimana merupakan suatu sistem untuk mengamankan jaringan nirkabel.

Sehingga tidak semua pengguna internet dapat mengakses langsung, maka ketika seorang pengguna yang ingin membuka situs internet, *chillispot* ini akan meminta pengguna untuk login terlebih dahulu, yaitu dengan meminta *username* dan *password*.

Chillispot sebagian besar digunakan oleh perusahaan dan universitas yang menginginkan untuk mengontrol akses yang masuk ke jaringan mereka. Fungsi utama *chillispot* adalah untuk memaksa *user* masuk ke halaman web khusus, umumnya untuk otentikasi. Kelemahan dari *chillispot* dibandingkan dengan yang lainnya, misalnya pada proses untuk melakukan otentikasi masih menggunakan *plain text* atau http. Sedangkan kelebihan dari *chillispot* itu sendiri yaitu adanya interkoneksi ke radius dan *database*, serta akses *user* berdasarkan waktu atau volume pemakaian.



2.5 Gambar Struktur Standar Chillispot^[6]

BAB III

PERANCANGAN & DESAIN SISTEM

3.1 Deskripsi Objek Penelitian.

3.1.1 Sejarah Singkat RT/RW NET

RT/RW-net secara teknologi, sama dengan warnet. Secara kasat mata, warnet menyambungkan sekumpulan komputer dalam sebuah ruangan/bangunan untuk disewakan. Sedangkan RT/RW-net menyambungkan sekumpulan komputer yang tersebar dalam sebuah RT atau RW. RT/RW-net adalah cara berbagi koneksi internet kerumah-rumah dalam satu RT/RW. Istilah RT/RW-net pertama kali digunakan sekitar tahun 1996 oleh para mahasiswa di Universitas Muhammadiyah Malang (UMM), seperti Nasar, Muji yang menyambungkan kos-kosan mereka ke kampus UMM yang tersambung ke jaringan A13 Indonesia melalui GlobalNet di Malang dengan gateway internet di ITB. Sambungan antara RT/RW-net di kos-kosan ke UMM dilakukan menggunakan walkie talkie di VHF band 2 meter pada kecepatan 1200bps.

Istilah RT/RW-net diutarakan oleh Bino, waktu itu tahun 1996 masih di GlobalNet, secara bercanda para mahasiswa Malang ini menamakan jaringan mereka RT/RW-net karena memang di sambungkan ke beberapa rumah di sekitar kos-kosan mereka.

3.1.2 Implementasi Awal RT/RW-net

Implementasi yang serius dari RT/RW-net dilakukan pertama kali oleh Michael Sunggiardi di perumahannya di Bogor sekitar tahun 2000-an. Banyak kisah sedih yang

diceritakan oleh Michael Sunggiardi karena sulitnya mencari pelanggan di awal 2-3 tahun operasi RT/RW-net-nya. Sebagian besar tetangga beliau pada saat itu tidak merasa butuh akses Internet 24 jam dari rumahnya.

Michael Sunggiardi banyak menggunakan kabel LAN untuk menyambungkan antar rumah. Karena lebih *reliable* dan lebih murah di bandingkan dengan menggunakan radio / *wireless* LAN.

3.2. Hasil Penelitian dan Pembahasan

3.2.1 Hasil Penelitian

Dari hasil peninjauan langsung di suatu kawasan RT/RW dengan majunya suatu teknologi terutama internet yang saat ini merupakan suatu kebutuhan bagi masyarakat luas dan sudah menjangkau kalangan menengah ke bawah. Dengan adanya jaringan internet yang cukup memadai saat ini, dan dengan pengembangan dan pemanfaatan jaringan internet via *Wireless* sudah bisa dimanfaatkan dengan baik, dan juga pemanfaatannya sudah dapat dirasakan oleh yang berada di area RT/RW yang terpasang *HostPot*.

Adanya jaringan *wireless* ini sudah bisa dimanfaatkan oleh para warga untuk menjelajahi dunia maya, informasi-informasi di dunia. Salah satu yang dapat diambil dari pemanfaatan jaringan ini adalah dengan adanya transaksi keuangan secara *online*, dan banyak yang lain lagi dimana memberikan manfaat ganda. Selain dapat menghemat waktu juga dapat membantu warga mengenal dan memanfaatkan fasilitas teknologi informasi yang disediakan oleh pihak RT/RW disuatu daerah.

Walaupun pemanfaatan jaringan internet via *wireless* sudah dapat di manfaatkan, namun pengelola dan *user* manajemen penggunaan dirasa kurang maksimal. Hal ini disebabkan oleh beberapa hal :

1. Setiap *user* yang akan menggunakan jaringan *wireless* tidak dapat terkontrol oleh pengelola jaringan di kawasan RT/RW ini.
2. Pengelola jaringan di kawasan RT.RW tidak mengetahui siapa saja yang sedang menggunakan fasilitas jaringan *Wireless* ini.
3. *User* yang ada jaringannya dapat menyalahgunakan fasilitas yang ada.
4. Pengelola jaringan di RT/RW net sering mengalami masalah pada koneksi jaringan yang sering terjadi, antara lain *Error conection*, *Ip conflic* dan masalah lainnya yang sering muncul.

Dengan adanya beberapa permasalahan di atas maka kiranya perlu dibangun manajemen sistem yang dapat mengatur, mengotrol dan mengelola pemanfaatan jaringan internet via *Wireless* ini. Manajemen sistem yang dimaksud adalah dengan dibuatnya sebuah *Hotspot Billing System*, dimana *user* yang akan menggunakan dan memanfaatkan koneksi ini bisa melakukan registrasi kepada pengelola jaringan yang ada dikawasan RT/RW net, sehingga pengelola dapat mengetahui siapa saja yang menggunakan fasilitas ini.

3.2.2 Perancangan Sistem

Dari hasil penelitian yang telah dilakukan maka dapat diketahui bahwa diperlukan sebuah *Hostpot Manajemen System* untuk mengatur, mengelolah dan

mengotrol fasilitas ini, karena sistem pengelola yang ada terasa belum maksimal. Sistem yang akan dirancang bertujuan untuk mempermudah dan juga menghemat biaya pemanfaatan jaringan internet dengan menggunakan kabel (*Wired*). Karena dengan menggunakan jaringan internet via nirkabel (*Wireless*) atau yang lebih sering disebut *HostPot* dapat mempermudah dan dapat meminimalisir biaya yang dikeluarkan bila masih menggunakan jaringan kabel (*wired*).

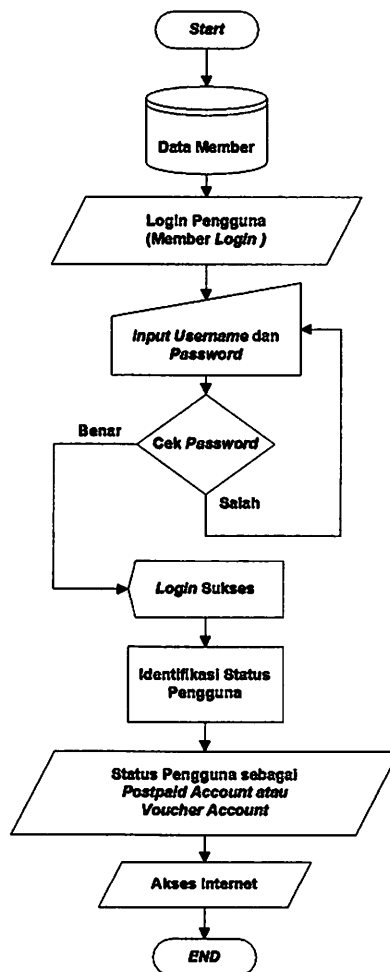
Pada sistem ini pengguna atau *user* yang akan menggunakan layanan ini harus melapor/ mendaftarkan diri terlebih dahulu kepada pengelola jaringan yang ada di suatu kawasan RT/RW net melalui SMS (*Sort Message Service*) atau bisa datang langsung kerumah pengelola. Tanpa melapor dan mendaftarkan diri ke pengelola pengguna atau user yang akan memanfaatkan fasilitas *hostpot* tidak akan mengakses internet.

Perancangan sistem ini menggunakan perangkat lunak Ubuntu 7.04 yang kemudian. Serta untuk *webserver* menggunakan *Aphace web server*, bahasa pemrograman yang digunakan untuk mendesain tampilan adalah PHP sebagai *compiler* dan *MySQL* (*phpMyAdmin*) sebagai *database*-nya. Adapun desain model pada perancangan system ini dapat digambarkan dalam bentuk *Logical Model* dan *Physical Model*. Bentuk *logical model* digambarkan dengan Diagram Berjenjang dan Diagram Arus Data (DAD). Sedangkan untuk bentuk *physical model* digambarkan dengan Bagan Alir (*flowchart*).

3.2.3 Bagan Alir *Billing System*

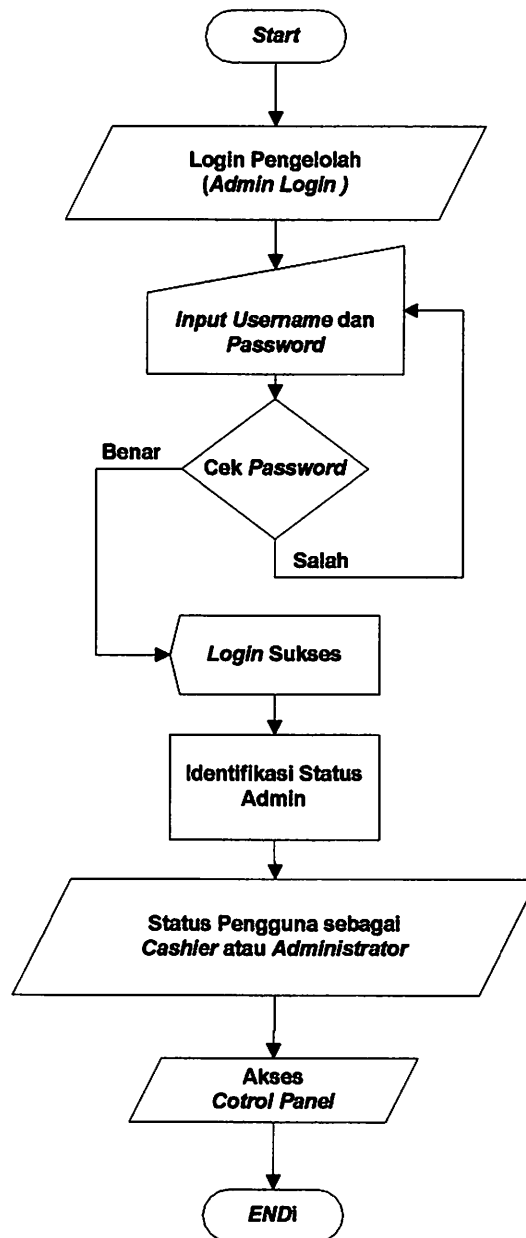
Dalam mengakses fasilitas-fasilitas *Billing System* baik pengguna maupun pengelola masing-masing mempunyai konten, *interface* dan proses yang berbeda-beda, untuk lebih jelasnya pembahasan mengenai hal ini akan dijelaskan melalui Bagan Alir secara terpisah

a. Bagan Alir Sistem Proses Login *User*



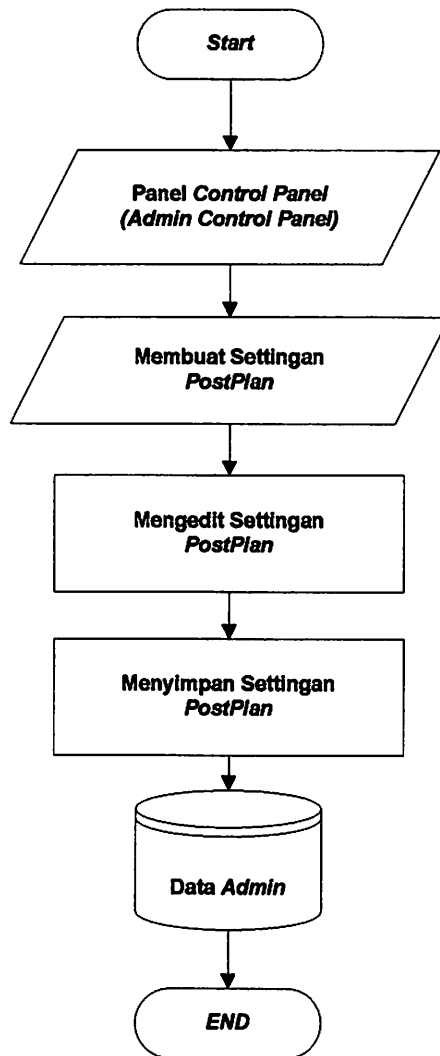
Gambar 3.1 Bagan Alir Sistem Proses Login *User*

b. Bagan Alir Sistem Login Pengelola



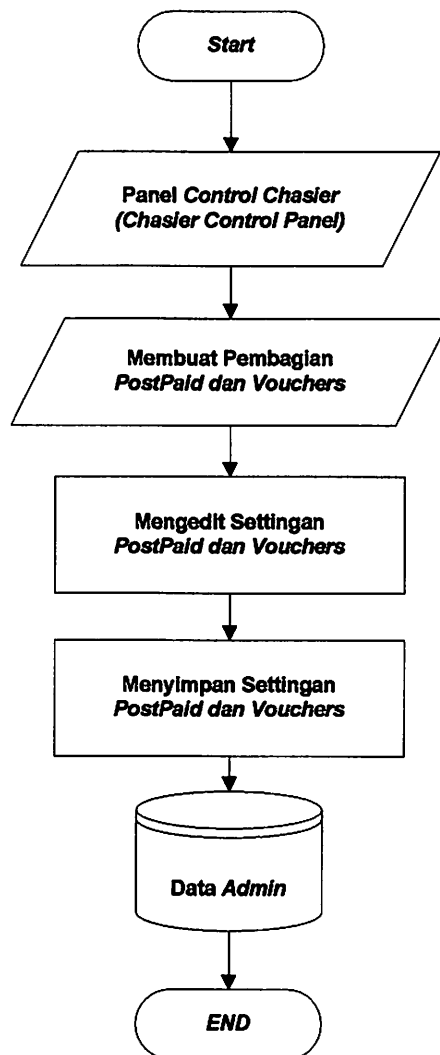
Gambar 3.2 Bagan Alir Sistem Login Pengelola

c. Bagan Alir Sistem *Control Panel PostPlan Administrator*



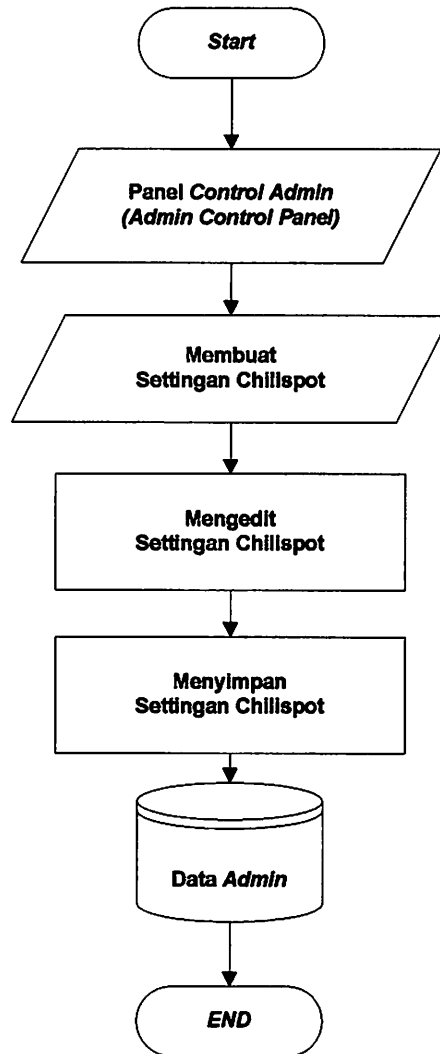
Gambar 3.3 Bagan Alir Sistem *Control Panel PostPlan Administrator*

d. Bagan Alir Sistem *Control Panel PostPlan Cashier*



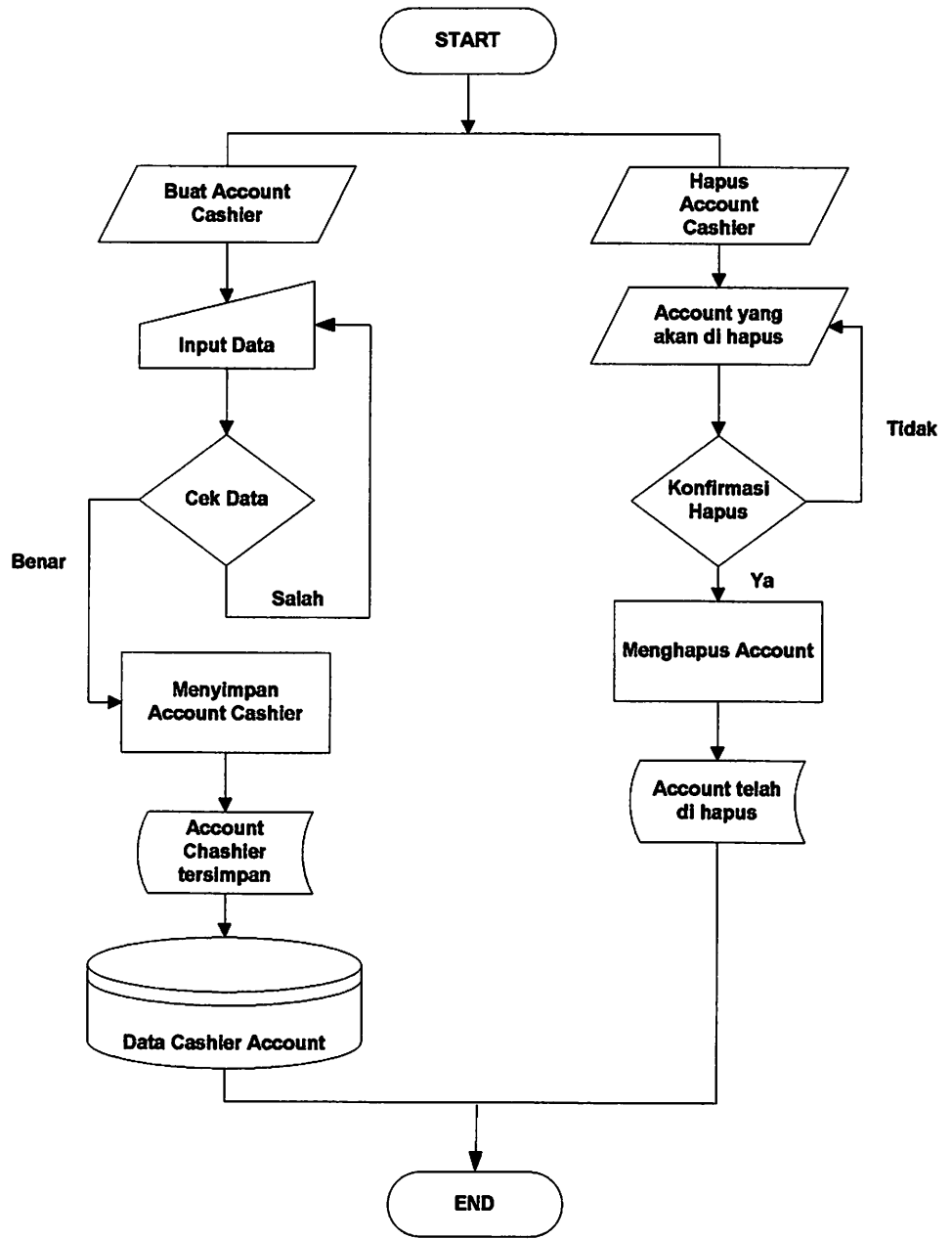
Gambar 3.4 Bagan Alir Sistem *Control Panel PostPlan Cashier*

e. Bagan Alir Sistem *Control Panel Chillisport Administrator*



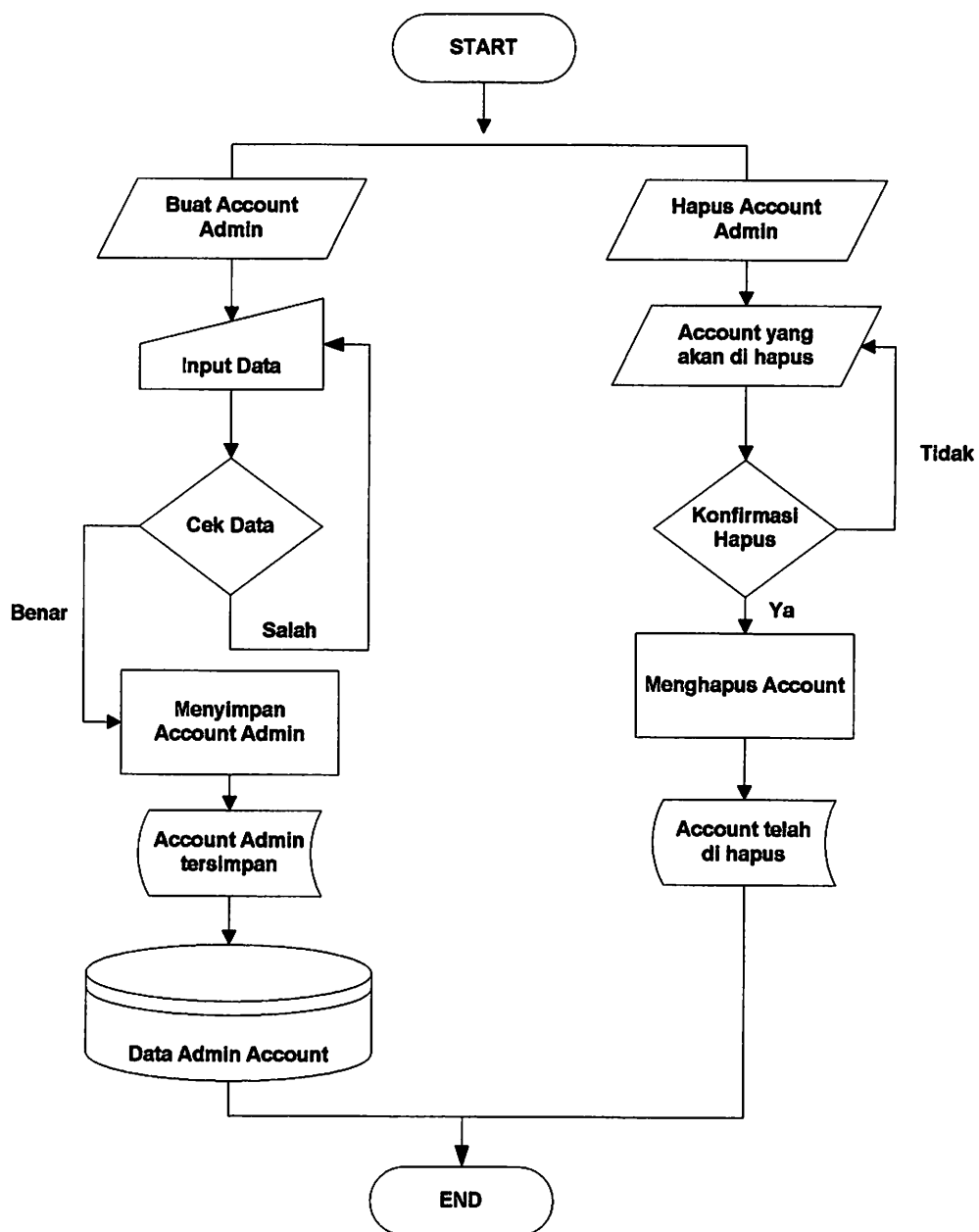
Gambar 3.5 Bagan Alir Sistem *Control Panel Chillisport Administrator*

f. Bagan Alir Sistem Control Panel Cashier Management Administrator



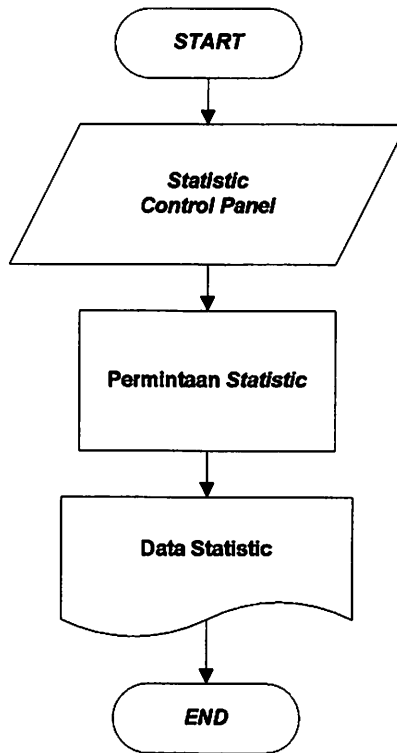
Gambar 3.6 Bagan Alir Sistem Control Panel Cashier Management Administrator

g. Bagan Alir Sistem Control Panel Security Management Administrator



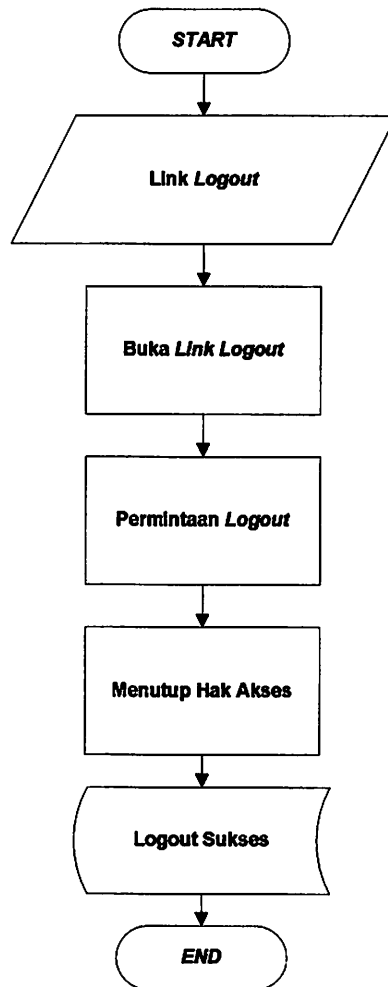
Gambar 3.7 Bagan Alir Sistem Control Panel Security Management Administrator

h. Bagan Alir Sistem *Statistic Management Administrator*



Gambar 3.8 Bagan Alir Sistem *Statistic Management Administrator*

i. Bagan Alir Sistem Proses *Logout*



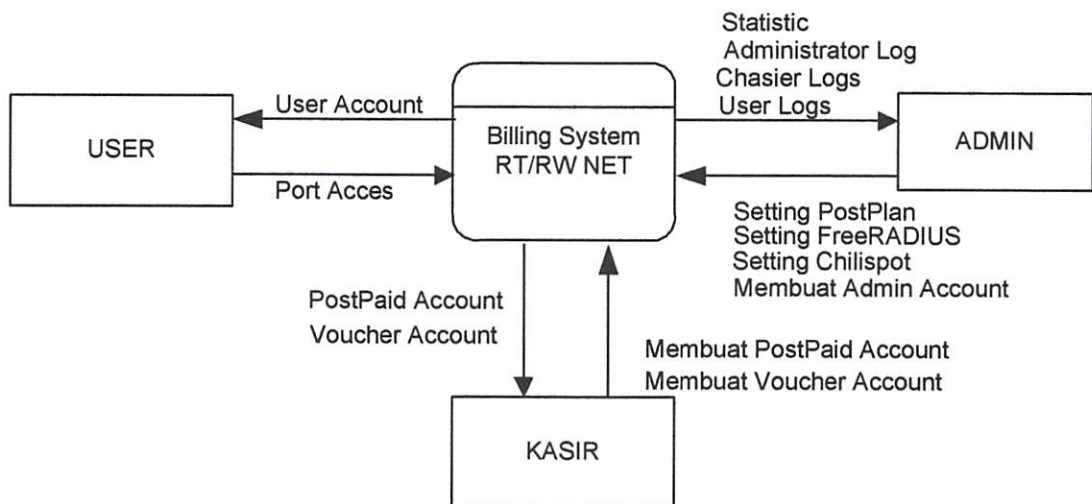
Gambar 3.9 Bagan Alir Sistem Proses *Logout*

3.2.4 Diagram Arus Data

Tabel 3.1 Identifikasi *Entity*

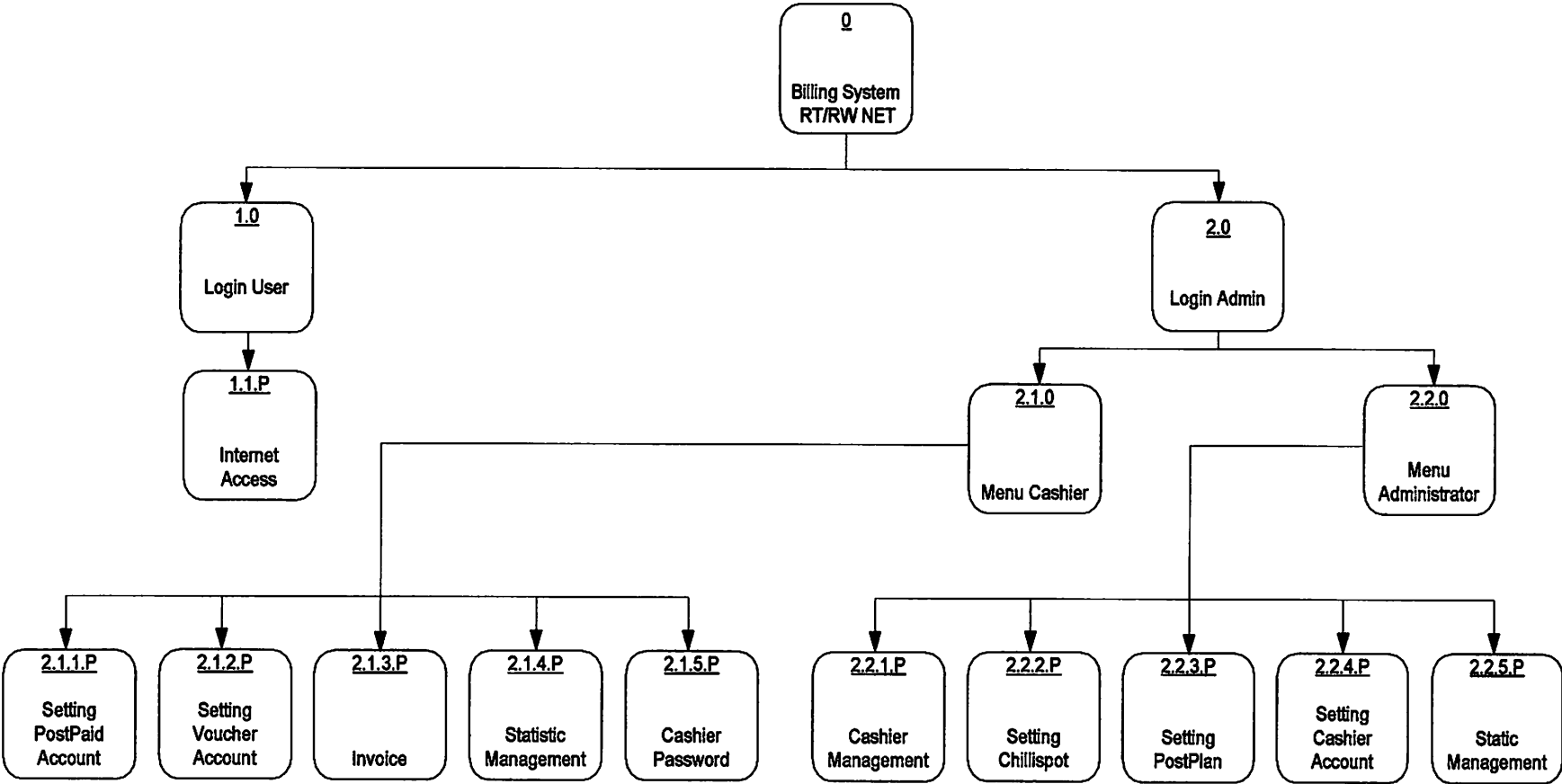
<i>EXTERNAL ENTITY</i>	<i>INPUT</i>	<i>OUTPUT</i>
<i>User</i>	<i>User Account</i>	<i>Port Acces</i>
<i>Cashier</i>	<i>Membuat PostPaid Account</i> <i>Membuat Voucher Account</i>	<i>PostPaid Account</i> <i>Voucher Account</i>
<i>Administrator</i>	<i>Setting PostPlan</i> <i>Setting FreeRADIUS</i> <i>Setting Chillisport</i> <i>Membuat Admin Account</i>	<i>Statistic</i> <i>Administrator Log</i> <i>Cashier Logs</i> <i>User logs</i>

a. Diagram Konteks



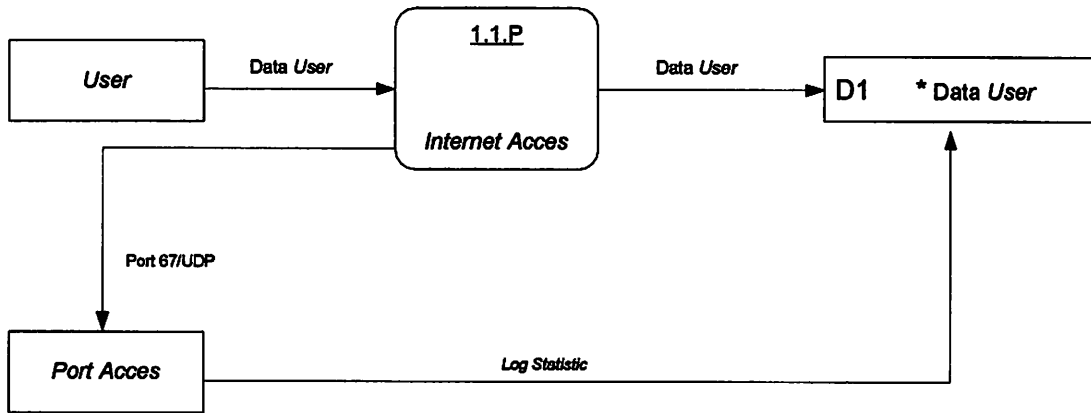
Gambar 3.10 Diagram Konteks Billing System RT/RW NET

b. Diagram berjenjang



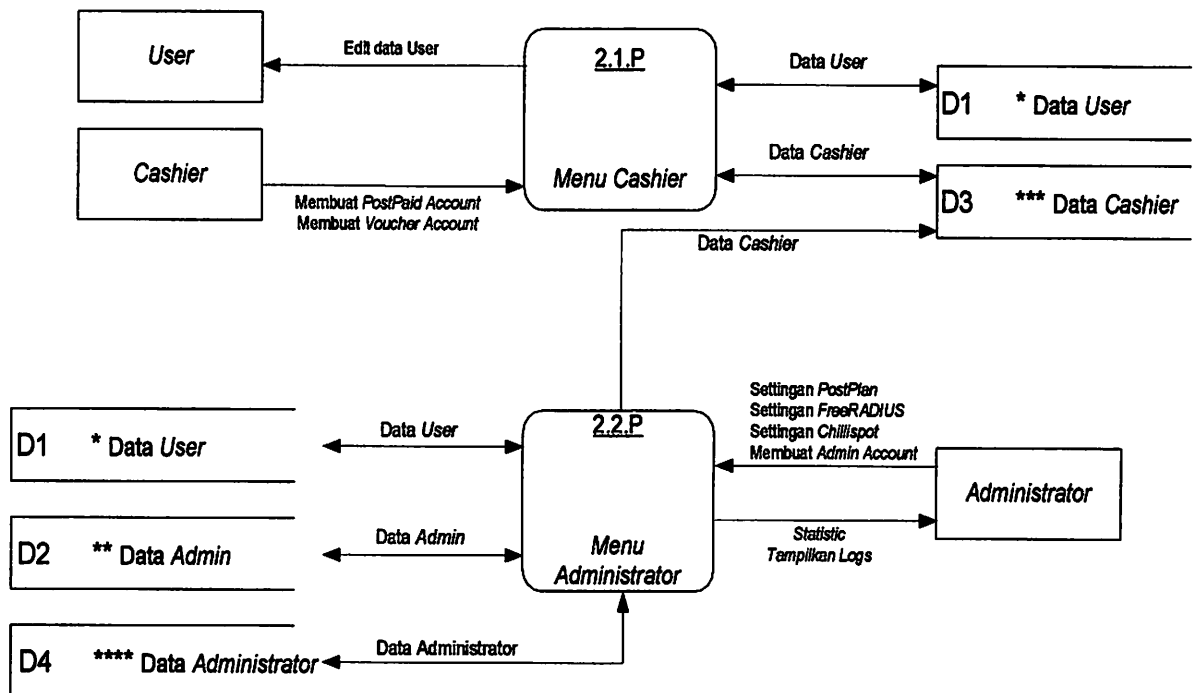
Gambar 3.11 Diagram berjenjang Billing System RT/RW NET

c. Diagram Arus Data Level 1 Proses 1



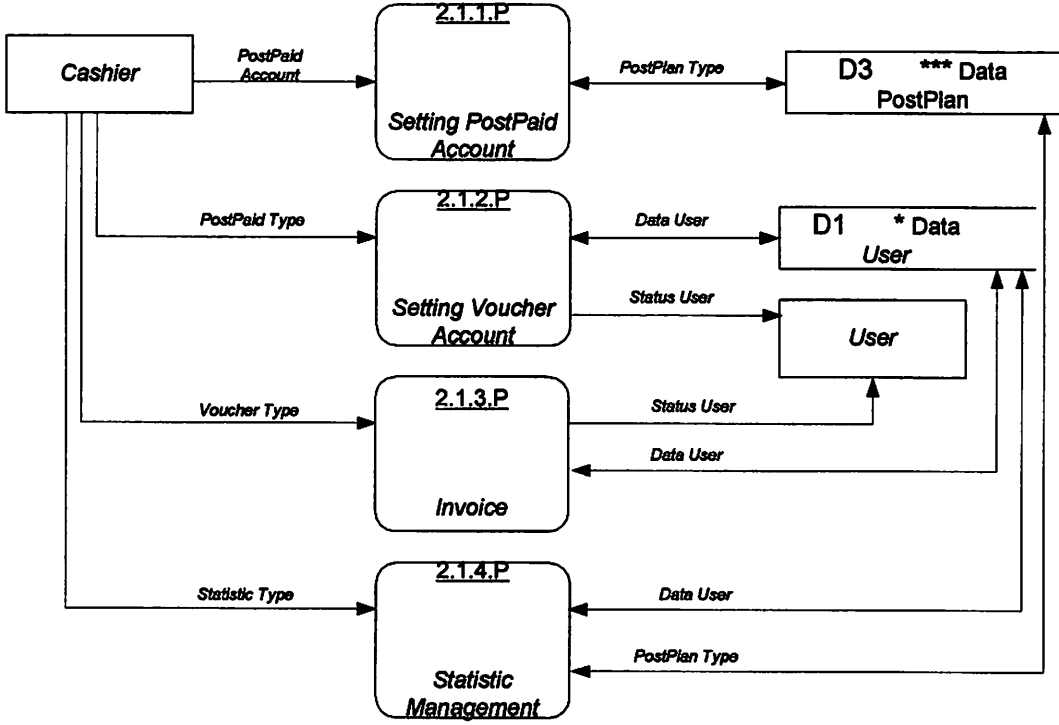
Gambar 3.12 DAD Level 1 Proses 1 Billing System RT/RW NET

d. Diagram Arus Data Level 1 Proses 2



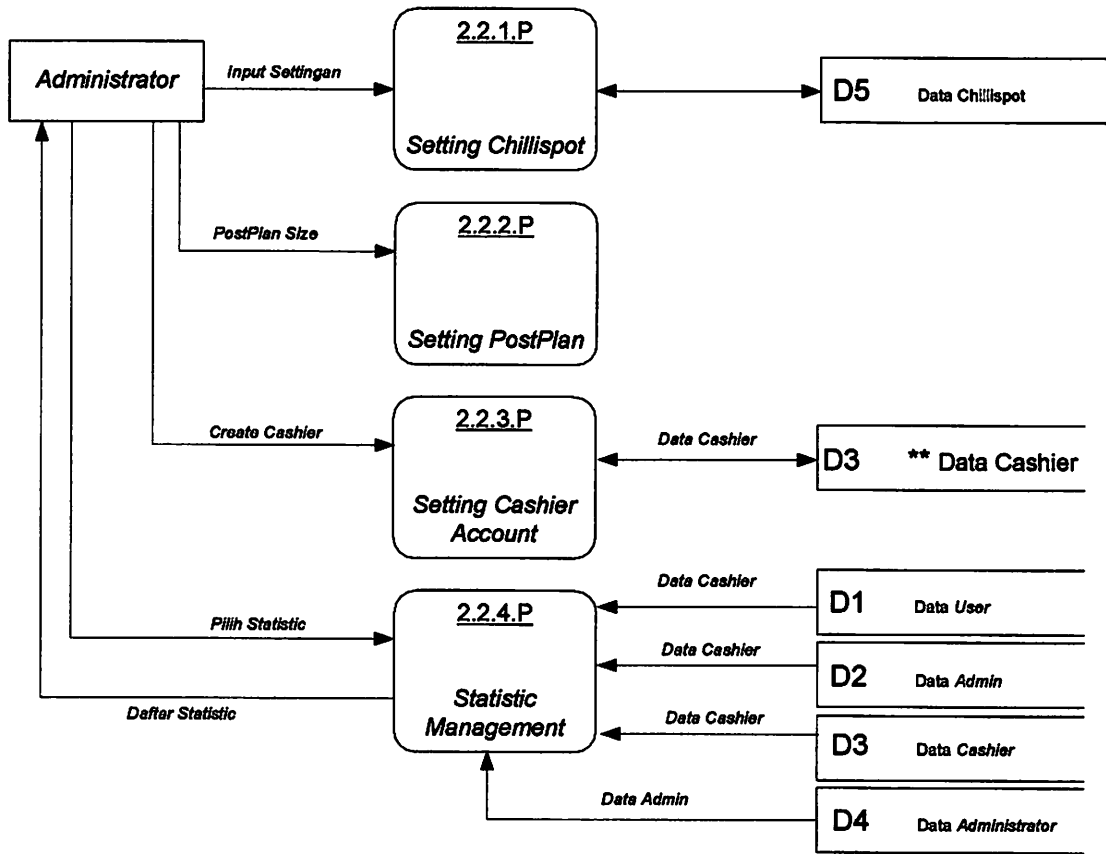
Gambar 3.13 DAD Level 1 Proses 2 Billing System RT/RW NET

e. Diagram Arus Data Level 2 Proses 1



Gambar 3.14 DAD Level 2 Proses 1 Billing System RT/RW NET

f. Diagram Arus Data Level 2 Proses 2



Gambar 3.15 DAD Level 2 Proses 2 Billing System RT/RW NET

3.2.5 Perancangan Database

Seperti yang telah di ketahui bahwa *Hotspot Billing System* ini dibuat dengan menggunakan *web server Apache*, dalam *web server* ini telah terdapat *file database* ini dengan ekstensi*.sql yang bias diakses dengan menggunakan phpMyAdmin (MySQL). *File database* inilah yang nantinya akan digunakan dalam perancangan database. Adapun tabel-tabel utama yang terdapat dalam *database* ini adalah sebagai berikut :

1. Tabel *Administrator Control Panel*

Tabel *Administrator Control Panel* ini terbagi atas :

a. Tabel Post Plan

Tabel 3.2 Struktur Tabel Post Plan

No	Nama Field	Tipe
1	Id	Int (255)
2	Name	Varchar (255)
3	Price	Int (9)

b. Tabel User Profile

Tabel 3.3 Struktur Tabel User Profile

No	Nama Field	Type
1	Id	Int (11)
2	Field 1	Varchar (50)
3	Field 2	Varchar (50)
4	Call me nicely	Varchar (3)

c. Tabel User

Tabel 3.4 Struktur Tabel User

No	Nama Field	Type
1	Id	Int (11)
2	User_name	Varchar (45)
3	Country_id	Int (11)
4	Password	Varchar (50)
5	Email	Varchar (120)
6	Role	Varchar (50)
7	Banned	Tinyint (1)
8	Forgotten_password_code	Varchar (50)
9	Last_visit	Datetime
10	Created	Timestamp
11	modified	Timestamp

Tabel-tabel di atas digunakan oleh Administrator untuk mengatur setingan penggunaan jaringan Hotspot. Pada tabel 3.2 Administrator mengatur pembagian penggunaan *Hotspot* yakni terdiri dari penggunaan perkuota *bandwith* (paket)per MB (*Mega Byte*) dan per-waktu (*time*)dalam menit. Dan selanjutnya 2 buah tabel digunakan untuk membuat *Account* Administrator dan juga *Account Cashier*.

2. Tabel *Cashier*

Memiliki lebih banyak tabel yang digunakan dikarenakan *Cashier* lebih banyak berperan dalam pembuatan *account* dan juga berbagai bentuk laporan penggunaan *Hotspot*. Tabel dapat dilihat sebagai berikut :

a. Tabel *Billing Plan*

Tabel 3.5 Tabel *Billing Plan*

No	Nama Field	Type
1	Id	Int(11)
2	Name	Varchar(255)
3	Type	Varchar(255)
4	Amount	Int(255)
5	Price	Float
6	idleTimeout	Int(255)
7	Simultaneous	Int(10)
8	Redirect_url	Varchar(255)
9	Bw_upload	Int(255)
10	Bw_download	Int(255)
11	Created_by	Varchar(255)

b. Tabel Invoice

Tabel 3.6 Tabel Invoice

No.	Nama Field	Type
1	Id	int (5)
2	Realname	varchar (255)
3	Username	varchar (255)
4	Used	int (255)
5	bill_by	varchar (255)
6	Date	Date
7	current_total	decimal (10,4)

c. Tabel Invoice Detail

Tabel 3.7 Tabel Invoice Detail

No.	Nama Field	Type
1	Id	int (11)
2	Realname	varchar (255)
3	Username	varchar (255)
4	Start	Datetime
5	Stop	Datetime
6	Used	int (11)
7	bill by	varchar(255)
8	Total	decimal (10,4)

d. Tabel *Postpaid Account*

Tabel 3.8 Tabel *Postpaid Account*

No.	Nama Field	Type
1	Id	int (255)
2	Realname	varchar (255)
3	Username	varchar (255)
4	Password	varchar (255)
5	bill by	varchar (255)
6	created by	varchar (255)

e. Tabel Postpaid Account Bill

Tabel 3.9 Tabel Postpaid Account Bill

No.	Nama Field	Type
1	Realname	Varchar (255)
2	Username	Varchar (255)
3	Password	Varchar (255)
4	Start	Datetime
5	Stop	Datetime
6	Time used	Decimal (14,4)
7	Packed used	Decimal (24,4)
8	Bill_by	Varchar (255)
9	Tune_price	Decimal (24,4)
10	Packet price	Decimal (34,4)

f. Tabel Postpaid Account List

Tabel 3.10 Tabel Postpaid Account List

No.	Nama Field	Type
1	Id	int (255)
2	Realname	varchar (255)
3	Username	varchar (255)
4	Password	varchar (255)
5	time used	decimal (36,4)
6	packet used	decimal (46,4)
7	bill_by	varchar (255)
8	time price	decimal (46,4)
9	packet price	decimal (56,4)

g. Tabel User Group

Tabel 3.11 tabel Postpaid Account List

No.	Nama Field	Type
1	Username	varchar (64)
2	groupname	varchar (64)
3	Priority	int (11)

h. Tabel Voucher

Tabel 3.12 Tabel Voucher

No.	Nama Field	Type
1	Id	int (255)
2	Username	varchar (255)
3	Password	varchar (255)
4	Billingplan	varchar (255)
5	Isprinted	tinyint (1)

i. Tabel Voucher List

Tabel 3.13 Tabel Voucher List

No.	Nama Field	Type
1	Id	int (255)
2	Username	varchar (255)
3	Password	varchar (255)
4	Billingplan	varchar (255)
5	Type	varchar (255)
6	Amount	int (255)
7	Price	Float
8	Time used	decimal (36,4)
9	Time remain	var (39)
10	Packet used	decimal (46,4)
11	Packet remain	var (49)
12	Isprinted	tinyint(1)
13	Valid	varchar (5)

Pada beberapa tabel di atas digunakan oleh *Cashier* antara lain:

- a. Pengaturan pembagian jenis *account* dan lamanya *account* aktif.
- b. Pengaturan *account user*.
- c. Detail laporan dari hasil pemakaian *account* yang ada.

3. Tabel *Chillispot*

Untuk *chillispot* hanya menggunakan 2 tabel sebagai penghubung *acces point* dengan *RADIUS Server* dan pencatatan *session chillispot*.

a. Tabel Chillispot Session

Tabel 3.1.4 Tabel Chillispot Session

No.	Nama Field	Type
1	session_id	varchar (40)
2	ip_address	varchar (16)
3	user_agent	varchar (50)
4	last_activity	int (10)
5	session_data	Text

b. Tabel NAS (*Network Acces Server*)

Tabel 3.15 Tabel NAS (*Network Acces Server*)

No.	Nama Field	Type
1	id	int (10)
2	nasname	varchar (128)
3	shortname	varchar (32)
4	type	varchar (30)
5	ports	int (5)
6	secret	varchar (60)
7	community	varchar (50)
8	description	varchar (200)

4. Tabel FreeRADIUS

Tabel *database* untuk FreeRADIUS digunakan sebagai pencatatan, perhitungan dan juga sebagai autentifikasi *user* atau pengguna jaringan *Hotspot*.

a. Tabel *Radius Account*

Tabel 3.16 Tabel Radius Account

No.	Nama field	Type
1	Radactid	Bigint (21)
2	Acctsessionid	Varchar(32)
3	Acctuniqueid	Varchar(32)
4	username	Varchar(64)
5	groupname	Varchar(64)
6	real	Varchar(64)
7	nasipaddress	Varchar(15)
8	nasportid	Varchar (15)
9	nasporttype	Varchar (32)
10	acctstarttime	Datetime
11	acctstoptime	Datetime
12	acctsesssiontime	Int (12)
13	acctauthentic	Varchar(32)
14	connectinfo_start	Varchar (50)
15	connectinfo_stop	Varchar (50)
16	acctinputoctets	Bigint (20)
17	acctoutputoctets	Bigint (20)
18	calledstationid	Varchar (50)
19	callingstationid	Varchar (50)
20	acctterminatecause	Varchar (32)
21	servicetype	Varchar (32)
22	framedprotocol	Varchar (32)
23	framedipaddress	Varchar (15)
24	acctstartdelay	Int (12)
25	acctstopdelay	Int (12)
26	xascendsessiosvrkey	Varchar (10)

b. Tabel *Radius Check*

Tabel 3.17 Tabel *Radius Check*

No.	Nama Field	Type
1	Id	Int(11)
2	Username	Varchar (64)
3	Attribute	Varchar (64)
4	Op	Char (2)
5	value	Varchar (253)

c. Tabel *Radius Group Check*

Table 3.18 Tabel Radius Group Check

No.	Nama Field	Type
1	Id	Int (11)
2	Groupname	Varchar (64)
3	Attribute	Varchar (32)
4	OP	Char (2)
5	Value	Varchar (253)

d. Tabel Radius Group Reply

Tabel 3.19 Tabel Radius Group Reply

No.	Nama Field	Type
1	Id	int (11)
2	Groupname	varchar (64)
3	Attribute	varchar (32)
4	Op	char (2)
5	value	varchar (253)

e. Tabel Radius Post Authorization

Tabel 3.20 Tabel Radius Post Authorization

No.	Nama Field	Type
1	Id	int (11)
2	Username	varchar (64)
3	Pass	varchar (64)
4	Reply	varchar (32)
5	Authdate	timestamp

f. Tabel Radius Reply

Tabel 3.21 Tabel Radius Reply

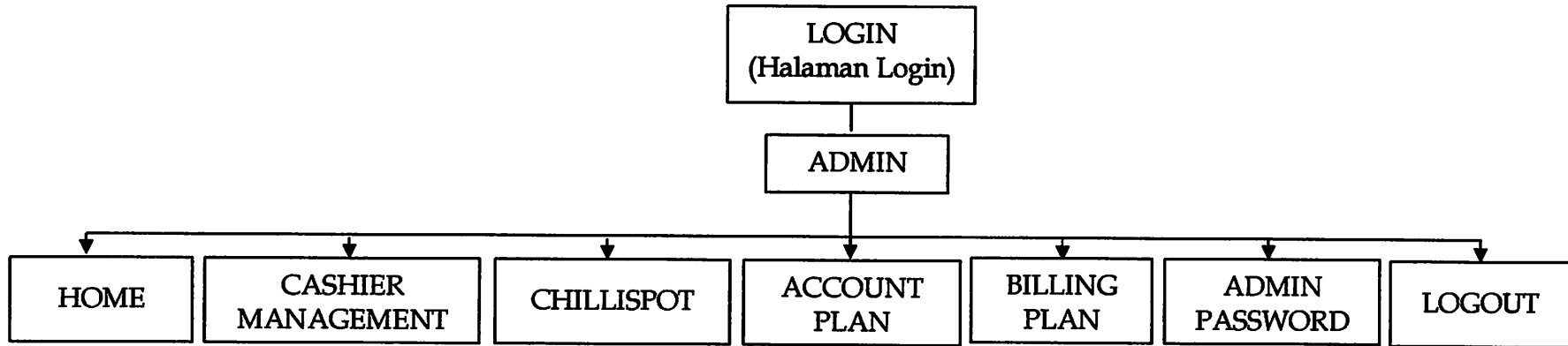
No.	Nama Field	Type
1	id	int (11)
2	username	varchar (64)
3	attribute	varchar (32)
4	op	char (2)
5	value	varchar (253)

3.2.6 Desain Billing System

Untuk membuat *Billing System* ini penulis membuatnya menggunakan *Open Source* Sistem. Yakni Ubuntu 7.04 yang kemudian penulis *remake* menjadi *easy hotspot* Dengan memanfaatkan kestabilan pengoperasian *server system* yang ada di sistem operasi. Ubuntu ini dibandingkan dengan distro linux yang lainnya. Serta menggunakan RADIUS server yakni FreeRADIUS dan juga sebagai pengontrol Acces Point digunakan Chillispot.

Desain *Billing Sistem* ini memiliki perbedaan halaman *Log In* antara *User* atau pengguna *hotspot* dengan Admin. Serta untuk Admin memiliki *Control Panel* yang berbeda antara Administrator dengan *Cashier*, hal ini dikarenakan fungsi antara Administrator dengan *Cashier* yang berbeda. Administrator lebih dikhususkan

pada pengelolaan jalannya sistem atau *System Control* sedangkan *Cashier* kepada pencatatan penggunaan *hotspot* dan pembuatan *Account User*, untuk lebih jelasnya desain *Billing System* ini dapat dilihat sebagai berikut:



Gambar 3.16 Desain Struktur Administrator

Keterangan :

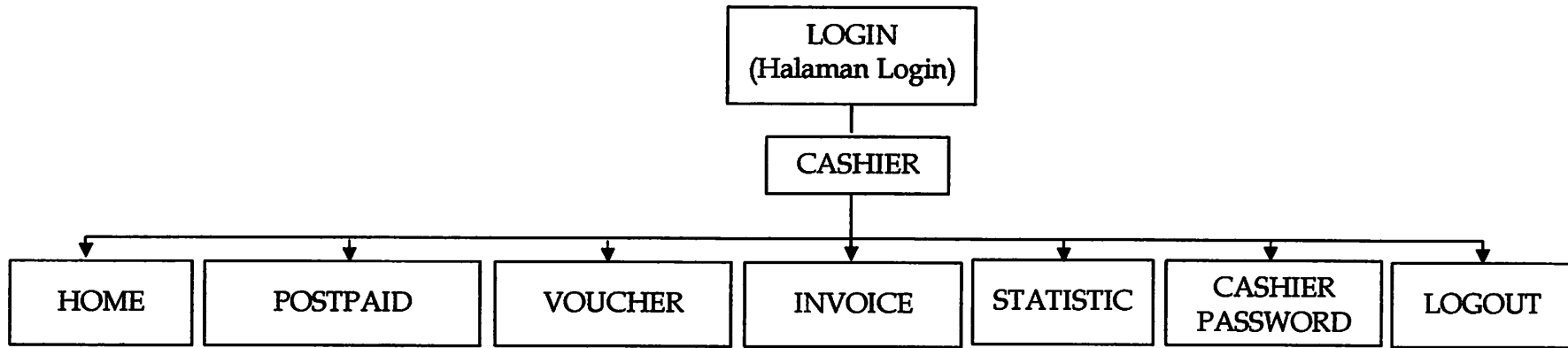
1. *Home*, merupakan halaman awal dari *GHOST Billing System*. Berisi informasi mengenai jenis organisasi, alamat dan juga informasi lainnya dari organisasi yang menggunakan *GHOST Billing System* ini serta sistem operasi yang digunakan pada komputer yang sedang mengaksesnya.
2. *Cashier Management*, merupakan halaman untuk membuat dan mengedit *Account Cashier*.
3. *Chillispot*, merupakan halaman untuk melakukan pengaturan pada *System Chillispot*

4. *Account Plan* merupakan halaman untuk mengatur kapasitas jenis akses yang di berikan pada *Hostpot*, yang terdiri atas 2 bagian, penggunaan berdasarkan waktu pemakain dan berdasarkan jumlah kapasitas *bandwith* yang digunakan.

5. *Billing Plan*, merupakan halaman dimana *administrator* dapat membagi lagi *Account Plan* menjadi beberapa bagian disesuaikan berdasarkan banyaknya *bandwith* yang digunakan atau berdasarkan berapa lama waktu yang dapat digunakan

6. *Admin Password*, merupakan halaman untuk membuat *account administrator* baru atau merubah *password* atau *username*.

7. *Logout* digunakan untuk keluar dari halaman adminstrator



Gambar 3.17 Struktur Kasir Desain

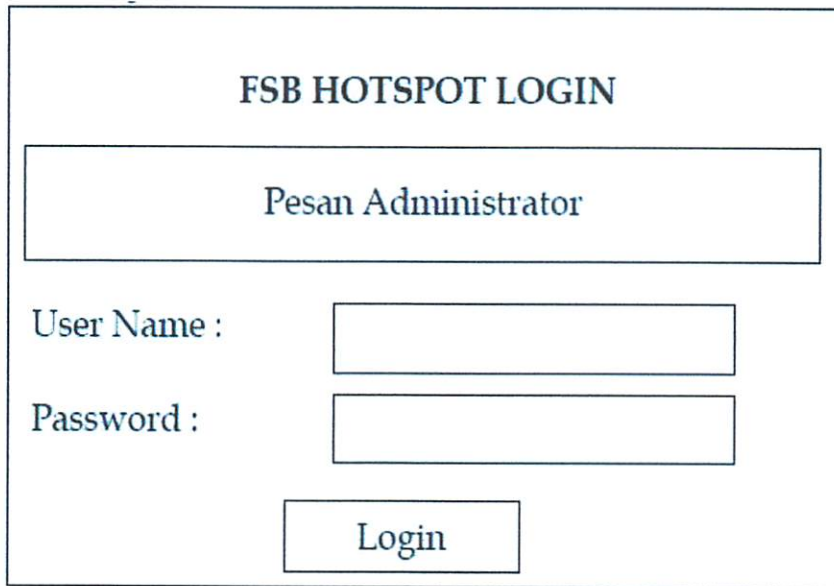
Keterangan :

1. *Home*, merupakan halaman awal dari *GHOST Billing System*. Berisi informasi mengenai jenis organisasi, alamat dan juga informasi lainnya dari organisasi yang menggunakan *GHOST Billing System* ini serta sistem operasi yang digunakan pada komputer yang sedang mengaksesnya
2. *Postpaid*, merupakan halaman untuk membuat *account user* dengan menentukan *username* dan *password* sendiri dan juga dapat merubah *account user* yang sudah pernah ada baik pada penggunaan berdasarkan waktu pemakaian ataupun kuota pemakaian.

3. *Voucher*, merupakan halaman untuk membuat *account voucher* yang dipergunakan sementara sampai batas waktu atau kuota yang ditentukan
4. *Invoice*, merupakan halaman yang digunakan untuk memeriksa penggunaan *account* yang pernah dicetak atau faktur pemakaian
5. *Statistic*, merupakan halaman untuk mengetahui berapa banyak *account* yang sudah pernah dibuat dan juga berapa persen penggunaan berdasarkan waktu ataupun berdasarkan kuota *bandwith* yang digunakan.
6. *Cashier password*, merupakan halaman untuk membuat *account cashier* baru atau merubah *password* atau *username*.
7. *Logout*, digunakan untuk keluar dari halaman *cashier*

3.2.7 Desain Input

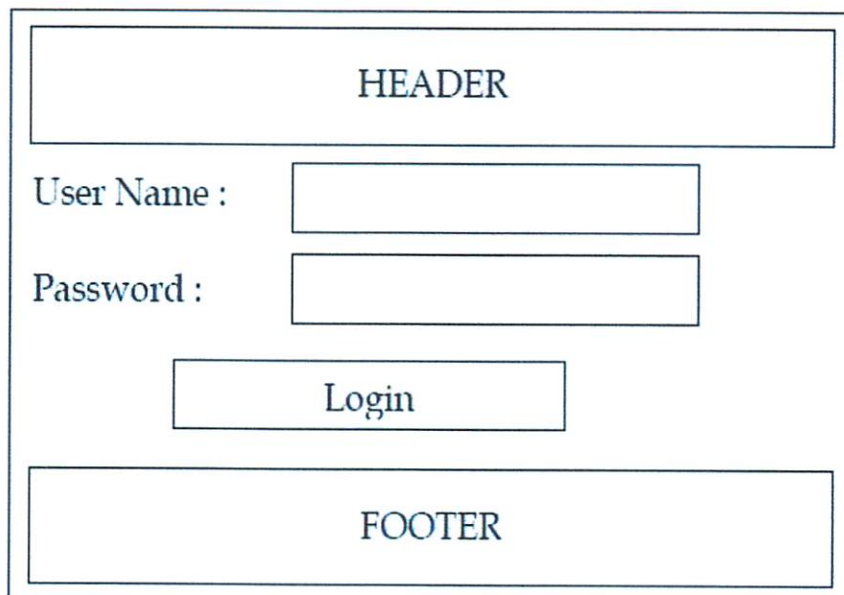
a. Desain Login User



The diagram shows a user login form titled "FSB HOTSPOT LOGIN". It features a header box labeled "Pesan Administrator". Below this, there are two input fields: "User Name :" and "Password :". A "Login" button is positioned below the password field.

Gambar 3.18 Desain Login User

b. Desain Login Admin



The diagram shows an admin login form. It has a "HEADER" box at the top and a "FOOTER" box at the bottom. In the center, there are two input fields: "User Name :" and "Password :". A "Login" button is located below the password field.

Gambar 3.19 Desain Login Admin

c. Desain Input Cashier Management

HEADER	
Icon 1 Icon 2 Icon 3 Icon 4 Icon 5 Icon 6 Icon 7	
Manage Cashiers	
Price/MB	<input type="text"/>
Price/Minute	<input type="text"/>
<input type="button" value="Save change"/>	
FOOTER	

Gambar 3.20 Desain Input Cashier Management

d. Desain Input PostPlan Administrator

HEADER	
Icon 1 Icon 2 Icon 3 Icon 4 Icon 5 Icon 6 Icon 7	
Postpaid Setting	
Price/MB	<input type="text"/>
Price/Minute	<input type="text"/>
<input type="button" value="Save change"/>	
FOOTER	

Gambar 3.21 Desain Input PostPlan Administrator

e. Desain Input BillingPlan

HEADER								
Icon 1 Icon 2 Icon 3 Icon 4 Icon 5 Icon 6 Icon 7								
Billing Plan								
Id	Name	Type	Amount	Price	DL Rate	Up Rate	Created By	
X	XXX	XXX	XXX	XXX	XXX	XXX	XXX	Icon8
X	XXX	XXX	XXX	XXX	XXX	XXX	XXX	Icon8
Name		<input style="width: 100%;" type="text"/>						
Type	<input style="width: 95%;" type="text"/> ▼							
Amount	<input style="width: 95%;" type="text"/>							
Price	<input style="width: 95%;" type="text"/>							
Download Rate	<input style="width: 95%;" type="text"/> ▼							
Upload Rate	<input style="width: 95%;" type="text"/> ▼							
Idle timeout	<input style="width: 95%;" type="text"/>							
<input type="button" value="Add Billing Plan"/>								
FOOTER								

Gambar 3.22 Desain Input BillingPlan

f. Desain Input Chillispot

HEADER	
Icon 1 Icon 2 Icon 3 Icon 4 Icon 5 Icon 6 Icon 7	
Internal captive portal management	
Radius server 1	<input style="width: 95%;" type="text"/>
Radius server 2	<input style="width: 95%;" type="text"/>
Radius secret	<input style="width: 95%;" type="text"/>
DHCP interface	<input style="width: 95%;" type="text"/>
UAM server	<input style="width: 95%;" type="text"/>
UAM secret	<input style="width: 95%;" type="text"/>
Client's homepage	<input style="width: 95%;" type="text"/>
Allow URL	<input style="width: 95%;" type="text"/> <small>separate by comma</small>
<input type="button" value="Save configuration"/>	
FOOTER	

Gambar 3.23 Desain Input Chillispot

g. Desain Input Postpaid Account

HEADER									
Icon 1 Icon 2 Icon 3 Icon 4 Icon 5 Icon 6 Icon 7									
Internal captive portal management									
Realname	Username	Password	Used	Bill by	Current total	Action			
xxx	xxx	xxx	xxx	xxx	xxx	Icon8	Icon9	Icon10	Icon11
xxx	xxx	xxx	xxx	xxx	xxx	Icon8	Icon9	Icon10	Icon11
Name	<input type="text"/>								
Username	<input type="text"/>								
Password	<input type="text"/>								
Bill by	<input type="text"/> ▼								
<input type="button" value="Add account"/>									
FOOTER									

Gambar 3.24 Desain Input Postpaid Account

h. Desain Input Add Cashier

HEADER	
Icon 1 Icon 2 Icon 3 Icon 4 Icon 5 Icon 6 Icon 7	
Add Cashier	
User profile	
Name	<input type="text"/>
Username	<input type="text"/>
Employed ID	<input type="text"/>
User main	
Username :	<input type="text"/>
e-mail :	<input type="text"/>
password :	<input type="text"/>
retype password :	<input type="text"/>
role :	<input type="text"/> ▼
is banned	<input type="checkbox"/>
<input type="button" value="add"/> <input type="button" value="reset"/>	
FOOTER	

Gambar 3.25 Desain Input Add Cashier

i. Desain Input Change Password Cashier

HEADER

Icon 1 | Icon 2 | Icon 3 | Icon 4 | Icon 5 | Icon 6 | Icon 7

Change Password

Username :

Old password :

New Password

Confirm :

FOOTER

Gambar 3.26 Desain Input Change Password Cashier

j. Desain Input Voucher Management

HEADER

Icon 1 | Icon 2 | Icon 3 | Icon 4 | Icon 5 | Icon 6 | Icon 7

Voucher Management

Icon 8

Username	Password	Billing plan	Time used	Time remain	Packet used	Packet remain	Icon9	Icon10	Icon11
XXX	XXX	XXX	XXX	XXX	XXX	XXX	Icon9	Icon10	Icon11
XXX	XXX	XXX	XXX	XXX	XXX	XXX	Icon9	Icon10	Icon11

Number of voucher:

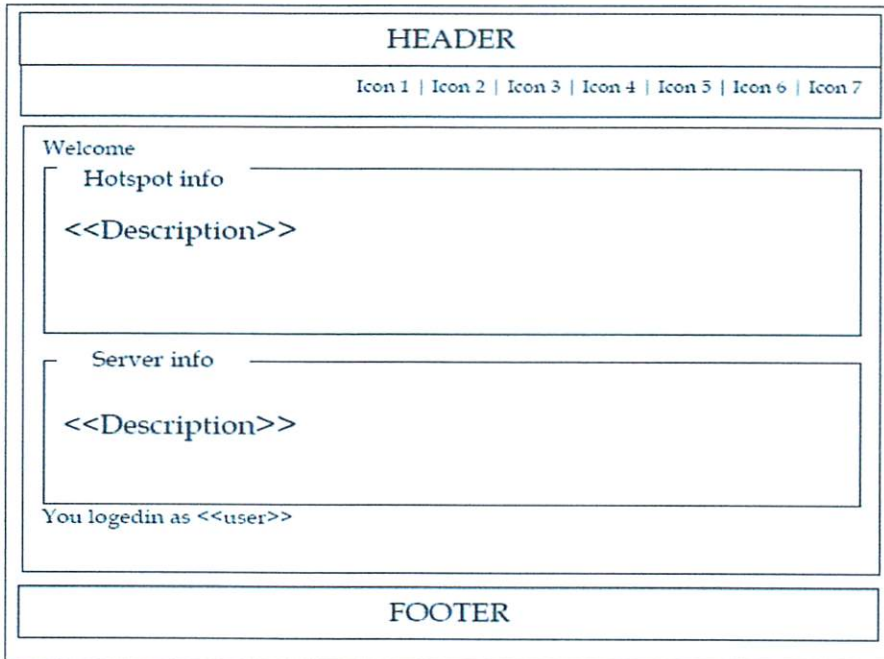
Billing Plan: ▼

FOOTER

Gambar 3.27 Desain Input Voucher Management

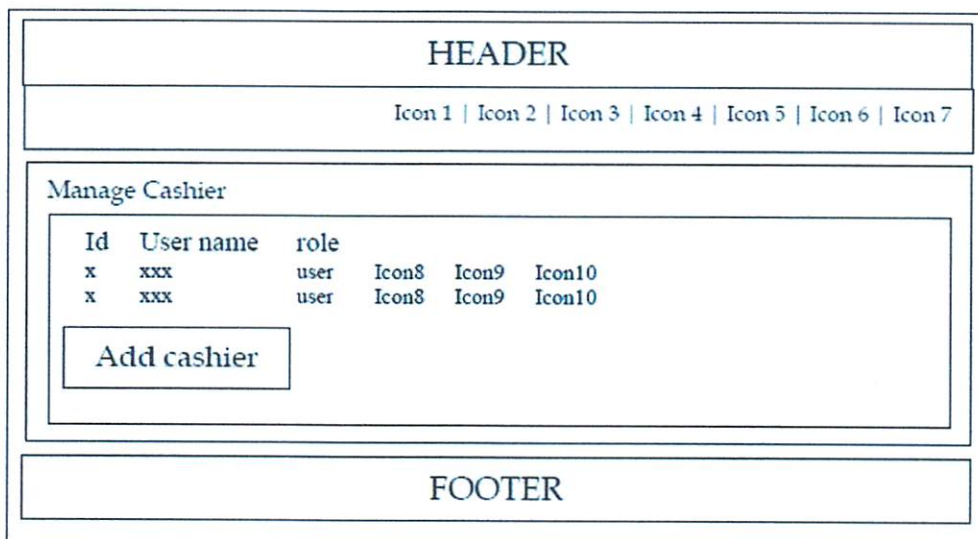
3.2.8 Desain Output

a. Desain Output Admin Control Panel Home



Gambar 3.28 Desain Output Admin Control Panel Home

b. Desain Output Manage Cashier



Gambar 3.29 Desain Output Manage Cashier

c. Desain Output Invoice

HEADER							
Icon 1 Icon 2 Icon 3 Icon 4 Icon 5 Icon 6 Icon 7							
Invoice Management							
Invoice no	Realname	Username	Used	Bill by	Date	Current total	detail
x	xxx	xxx	xxx	xxx	xxx	xxx	Icon8
x	xxx	xxx	xxx	xxx	xxx	xxx	Icon8
FOOTER							

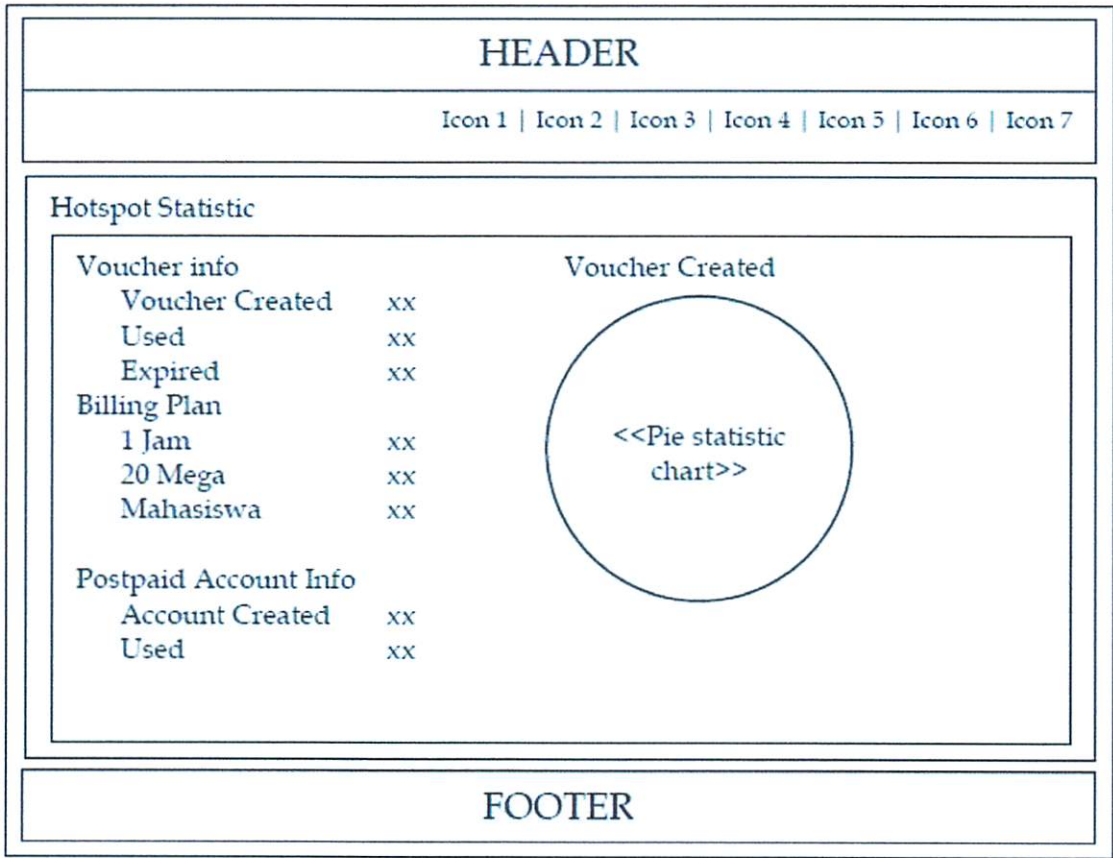
Gambar 3.30 Desain Output Invoice

d. Desain Output Logout User



Gambar 3.31 Desain Output Logout User

e. Desain Output Statistic



Gambar 3.32 Desain Output Statistic

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi

4.1.1 Instalasi OS (*Operating System*)

Karena *Hotspot Billing System* yang akan digunakan ini berbasis *Open Source* maka langkah awal yang dilakukan adalah memilih dan melakukan instalasi sistem operasi. Setelah dipilih berdasarkan beberapa distro linux yang ada maka akan digunakan ubuntu 7.04 sebagai sistem operasi *Hotspot Billing System*. Bagi anda yang pertama kali menggunakan Linux tidak perlu khawatir akan konfigurasi instalasi ataupun soal tata bahasa yang akan digunakan pada proses instalasi ini, dikarenakan Linux telah mendukung prosedur instalasi dengan bahasa indonesia.

4.1.2 Persiapan *Hardware Server*

Untuk membuat sebuah *Hotspot Billing System* ini dibutuhkan beberapa *Hardware* sebagai berikut :

- a. PC Intel Pentium IV atau lebih baru
- b. RAM 1028 Mb
- c. Haddisk 40GB atau lebih besar (Kebutuhan OS 1,5 GB)
- d. Acces Point (Model apa saja, yang dibutuhkan yang mendukung DHCP)
- e. Network Interface Card (LAN Card)
- f. Switch (optional)

4.1.3 *Install ubuntu*

Setelah *hardware* PC-nya telah disiapkan saatnya untuk proses instalasi Xubuntu. Pada intinya untuk proses instalasi Edubuntu/Ubuntu/Xubuntu tidak jauh berbeda, pertama-tama masukkan CD ubuntu yang telah dibuat sebelumnya dan pastikan PC anda membaca CD-Room saat booting.. lakukan langkah –langkah instalasi ubuntu sesuai petunjuk hingga berhasil.

4.1.4 *Proses Remake dari Ubuntu ke Distro RT/RW NET Hostpot*

Proses *remake* dari system operasi ubuntu ke sistem operasi distro linux baru yakni RT/RW NET *Hostpot* ini membutuhkan keahlian dan ketelitian. Selain itu dibutuhkan waktu yang tidak sedikit. Namun dikarenakan distro linux merupakan OSS (*Open Source Software*) maka kita dengan mudah memperoleh paduan materi proses *Remake* ini dari internet dan juga buku-buku Linux yang sudah banyak beredar. Langkah-langkah untuk membuat distro linux yang baru adalah sebagai berikut :

4.1.4.1 *Software Reqriutment*

Sebagai persiapan membuat distro baru beberapa *software* yang harus disiapkan antara lain :

- a. *Chillispot*
- b. *RADIUS Server (FreeRADIUS)*
- c. *Web Server (Apache)*
- d. *Database (MySQL)*
- e. *Reconstructor*

4.1.4.2 System Configurasi

Langkah awal yang harus dilakukan dalam persiapan *server* ini kita harus melakukan settingan terhadap sistem ubuntu sebelum dirubah ke *Distro Hostpot* ini. Untuk itu langkah-langkah *settingan system* sebagai berikut :

a) Setting account root

- a. Langkah awal adalah masuk ke sistem dengan menggunakan *username* dan *password* yang telah dibuat saat instalasi system
- b. Karena saat ini status kita hanya sebagai *user* maka kita tidak dapat melakukan konfigurasi langsung terhadap *file* sistem, oleh sebab itu kita perlu masuk sebagai *root* dalam hal ini *root* adalah *account* administrator.
- c. Buka terminal program kemudian ketik “*sudo su -*”, selanjutnya masukkan *password user* yang kita gunakan. Perbedaan yang akan tampak dari perubahan *user* ke *root* adalah tanda akhir pada terminal program yakni jika status kita *user* maka yang akan muncul adalah tanda “\$” sedangkan bila kita sebagai *root* akan muncul tanda “#”.
- d. Selanjutnya adalah melakukan settingan *password* untuk *root* dengan cara, ketik “*password root*” kemudian tekan “*enter*”, selanjutnya akan diminta memasukan *password* untuk *account root* yang akan kita gunakan dan masukan sekali lagi sebagai konfirmasi *password root*.

b) Network Interface Configuration (NIC)

Untuk pengaturan NIC yang kita gunakan adalah *Ethernet Card*. Sebenarnya untuk melakukan settingan pada eth 0 bisa kita lakukan langsung dari *interface Xubuntu* pada setinggan *connection*. Fungsi dari eth 0 adalah untuk koneksi langsung ke

koneksi internet ataupun koneksi ke *router server*. Untuk saat ini kita akan melakukan setingan pada *eth1* yang akan dihubungkan langsung ke *acces point* atau bisa kita gunakan *switch* bila akan menggunakan beberapa *acces point* sekaligus. Adapun langkah-langkahnya adalah sebagai berikut :

- a. Buka terminal program kemudian ketik “*nano/etc/network/interfaces*”.

Setelah itu akan muncul tampilan teks di jendela terminal program dengan setingan utama pada akhir *interface configuration* sebagai berikut :

--isi tulisan ini bagian paling bawah dari *interfaces*--

```
# The secondary network interface
```

```
auto eth1
```

```
iface eth1 inet dhcp
```

--berakhir di atas tulisan ini--

- b. Selanjutnya tekan “*Crt+X*” kemudian ketik “*Y*” kemudian tekan “*enter*” untuk menyimpan konfigurasi yang telah kita lakukan. Alasan menggunakan *DHCP configuration* pada *eth1* ini karena akan terhubung dengan *chillispot* yang akan memberikan *ip address*.

- c. Selanjutnya adalah melakukan settingan DNS yang akan digunakan sesuai dengan DNS yang diberikan oleh ISP yang kita gunakan.

Ketik “*nano /etc/resolv.conf*”

-- isi dengan perintah berikut dibawah --

```
nameserver 202.155.0.10
```

```
nameserver 202.155.0.15
```

-- berakhir diatas dari tulisan ini --

simpan dengan menekan “ctrl+x” tekan “y” kemudian enter

contoh diatas adalah permisalan DNS 1 dan DNS 2 nya.

- d. Selanjutnya adalah edit file sysctl.conf

```
#nano /etc/sysctl.conf
```

cari tulisan berikut:

```
# net.ipv4.conf.default.forwarding=1
```

hilangkan tanda pagar nya jadi:

```
net.ipv4.conf.default.forwarding=1
```

simpan dan keluar dengan perintah: “ctrl+x”, ketik “y” dan tekan “enter”.

- e. Setting rc.local

```
#nano /etc/rc.local
```

hapus exit=0

-- masukan tulisan berikut dibagian paling bawah rc.local —

```
echo \"1\" > /proc/sys/net/ipv4/ip_forward
```

-- berakhir diatas tulisan ini –

simpan dan keluar dengan perintah: ctrl+x, ketik y dan tekan enter

- f. Selanjutnya adalah proses reboot untuk mengaktifkan semua setingan kita tadi dengan mengetikkan perintah “reboot” pada terminal program.

4.1.4.3 Program and Support Installation

Langkah selanjutnya adalah melakukan proses instalasi program utama dan program pendukung untuk pembuatan *server hotspot*. Adapun langkah-langkahnya adalah sebagai berikut :

a. Install program – program yang dibutuhkan :

```
#apt-get install unzip
```

```
#apt-get install fakeroot
```

```
#apt-get install ssh
```

```
#apt-get install build-essential
```

```
#apt-get install rrdtool
```

```
#apt-get install snmp
```

```
#apt-get install mysql-server
```

```
#apt-get install apache2 php5 php5-mysql
```

```
#apt-get install ssl-cert
```

```
#apt-get install freeradius freeradius-mysql
```

```
#wget http://www.chillispot.info/download/chillispot_1.0_i386.deb
```

```
#dpkg -i chillispot_1.0_i386.deb
```

Ket:

Sewaktu menginstall mysql-server anda akan dimintai *password* untuk root mysql.

Jika anda tidak dimintai untuk memasukkan *password* anda dapat membuatnya dengan perintah:

```
#mysqladmin -u root password 'easyhotspot'
```

b. buat file options di folder network dengan perintah :

```
nano /etc/network/options
```

```
-- tulis sebagai berikut --
```

```
ip_forward = yes
```

```
-- berakhir diatas tulisan ini --
```

simpan dan keluar dengan perintah:

ctrl+x, ketik y dan tekan enter

restart network dengan perintah

```
#/etc/init.d/networking restart
```

c. Edit file apache2.conf

```
#nano /etc/apache2/apache2.conf
```

-- dibagian paling bawah nya ditulis --

```
ServerName 192.168.10.10
```

-- berakhir diatas tulisan ini --

Simpan make perintah:

Ctrl+x ketik y dan tekan enter

ip address 192.168.10.10 ini tergantung dari

eth0 ip anda

Restart apache2 nya:

```
#/etc/init.d/apache2 restart
```

d. Edit file php.ini

```
#nano /etc/php5/apache2/php.ini
```

Cari tulisan:

```
; extension=mysql.so
```

Hilangkan tanda titik koma nya dan ganti mysql.so menjadi mysql.so

```
extension=mysql.so
```

Cari juga tulisan berikut:

```
post_max_size = 8M
```

Ubah 8M menjadi 16 M:

```
post_max_size = 16M
```

Simpan dengan perintah:

Ctrl+x ketik y kemudian enter

e. ambil paket phpmyprepaid dan letakkan di folder /var/www dan ekstrak

```
#cd /var/www
```

```
#rm -rf apache2-default
```

```
#wget
```

```
http://downloads.sourceforge.net/phpmyprepaid/phpmyprepaid04b
```

```
3.tgz?modtime=1187550638&big_mirror=0
```

```
#tar -zxvf phpmyprepaid04b3.tgz
```

g. install program phpmyprepaid

Untuk meng-*install* phpmyprepaid, anda perlu menggunakan komputer desktop lain dalam satu jaringan yang ter-*install* mozilla firefox. Alasannya karena ada beberapa tampilan yang tidak bisa ditampilkan di web browser lain selain mozilla.

Satu lagi, nyettingnya make ssh .

Buka mozilla firefox dan arahkan addressnya ke :

```
http://192.168.10.10/phpmyprepaid/www/install/setup.php
```

perlu di ingat ip address 192.168.10.10 ini hanya permisalan saja, ini tergantung dari settingan eth0 ip address anda.

Kik start.

Pilih I accept dan klik next lagi.

Perhatikan dan edit konfigurasi seperti berikut:

hpMyPrepaid install directory : /var/www/phpmyprepaid

FreeRADIUS binary files directory : /usr/sbin

FreeRADIUS config files directory : /etc/freeradius

FreeRADIUS Dictionary directory : /usr/share/freeradius

FreeRADIUS start/stop/restart/status script : /etc/init.d/freeradius

RRDTOOL binary path : /usr/bin/rrdtool

Sudo binary path : /usr/bin/sudo

System log file path : /var/log/messages

FreeRADIUS radius.log file path : /var/log/freeradius/radius.log

Radclient command : /usr/bin/radclient

MySQL client command : /usr/bin/mysql

snmpwalk command : /usr/bin/snmpwalk

snmpget command : /usr/bin/snmpget

Klik next

Disini anda diminta untuk mem verifikasi konfigurasi.. jika ada yang belum periksa kembali apa yang kurang, dan klik recheck.. jika sudah klik next.

Pada bagian ini anda akan diminta untuk mengkonfigurasi *database*, anda dapat mencontoh sebagai berikut:

Root password for Mysql : sangatrahasia

PhpMyPrepaid Database Name (phpmyprepaid) : phpmyprepaid

PhpMyPrepaid Database Password : sangatrahasia

Confirm Password : sangatrahasia

Database location (localhost) : localhost

FreeRADIUS location (localhost). *Not applicable by now : localhost

FreeRADIUS version. *Not applicable by now : 1.X

Klik next

Pada bagian ini anda akan diminta untuk mengkonfigurasi user interface yaitu sebagai super administrator dalam manajemen phpmyprepaid, anda dapat mencontoh sebagai berikut:

Administrator login for PhpMyPrepaid : indra

Administrator password for PhpMyPrepaid : sangatrahasia

Confirm Password : sangatrahasia

Administrator name for PhpMyPrepaid : indra

Administrator surname for PhpMyPrepaid : indra

Administrator email for PhpMyPrepaid :rahadian_indra@yahoo.com

This e-mail address is being protected from spam bots, you need

JavaScript enabled to view it

Administrator language for PhpMyPrepaid. *Only english is supported

by now. : es

Klik next

Pada bagian ini merupakan informasi yang dilaporkan oleh phpmyprepaid dalam membuat database pada mysql

Klik next

Anda akan diminta untuk memasukkan lokasi daerah anda, contohnya sebagai berikut:

Location name : Mergan Raya

Location Domain : `www.merganraya.co.cc`

Location Country Code : Indonesia

Location Area code : 96128

Location City : Malang

Klik next

Instalasi finish, silahkan klik interface

Pertama anda diminta untuk menghapus folder install dan mengubah hak akses folder `www` menjadi `755`.

```
#rm -rf /var/www/phpmy prepaid/www/install
```

```
#chmod 755 /var/www/phpmy prepaid/www
```

g. Setting chillispot

a. edit file `chilli.conf`

```
#nano /etc/chilli.conf
```

-- contoh yang harus disetting --

```
net 192.168.2.0/24
```

```
dns1 202.93.32.146
```

```
dns2 202.93.32.147
```

```
radiusserver1 127.0.0.1
```

```
radiusserver2 127.0.0.1
```

```
radiussecret sangatrahasia
```

```
dhcpif eth1
```

```
uamserver https://192.168.100.1/cgi-bin/hotspotlogin.cgi
```

```
uamsecret sangatrahasia
```

uamlisten 192.168.2.15

-- berakhir diatas tulisan ini --

Simpan make perintah:

Ctrl+x tekan y kemudian enter

Ket:

- uamlisten 192.168.2.15 disini adalah ip address eth1 yang akan dimasukkan oleh

Chillispot

- net 192.168.2.0/24 adalah ip address dhcp yang diberikan oleh chillispot kepada user, mulai dari 192.168.2.2 – 192.168.2.254

b. Buat file hotspotlogin.cgi di folder cgi-bin

```
#cp /usr/share/doc/chillispot/hotspotlogin.cgi.gz /usr/lib/cgi-bin
```

```
#gunzip /usr/lib/cgi-bin/hotspotlogin.cgi.gz
```

```
#cd /usr/lib/cgi-bin/
```

```
#chmod 755 hotspotlogin.cgi
```

```
#nano hotspotlogin.cgi
```

-- cari tulisan berikut --

```
# $uamsecret = \"sangatrahasia\";
```

Dan

```
# $userpassword=1;
```

Hilangkan tanda pagarnya menjadi:

```
$uamsecret = \"sangatrahasia\";
```

Dan

```
$userpassword=1;
```

Simpan make perintah:

Ctrl+x tekan y kemudian enter

c. Aktifkan firewall chillispot dan buat firewallnya agar diaktifkan setiap kali server restart

```
#sh /usr/share/doc/chillispot/firewall.iptables
```

```
#cp/usr/share/doc/chillispot/firewall.iptables/etc/init.d/chilli.ip
```

Tables

```
#chmod u+x /etc/init.d/chilli.iptables
```

```
#ln -s /etc/init.d/chilli.iptables /etc/rcS.d/S40chilli.iptables
```

h. Setting freeradius

a. Edit file clients.conf

```
#nano /etc/freeradius/clients.conf
```

-- cari konfigurasi berikut dan modifikasi menjadi seperti ini --

```
client 127.0.0.1 {
```

```
secret = sangatrahasia
```

```
shortname = localhost
```

```
nastype = other
```

```
}
```

-- berakhir diatas tulisan ini --

Simpan make perintah:

Ctrl+x tekan y kemudian enter

b. Setting file sql.conf

```
#nano /etc/freeradius/sql.conf  
  
-- cari konfigurasi berikut dan edit --  
  
# Connect info  
  
server = \"localhost\"  
  
login = \"phpmy prepaid\"  
  
password = \"sangatrahasia\"  
  
# Database table configuration  
  
radius_db = \"phpmy prepaid\"
```

c. edit file radiusd.conf

```
#nano /etc/freeradius/radiusd.conf  
  
-- cari konfigurasi berikut --  
  
\"$INCLUDE ${confdir}/sql.conf\"  
  
tambahkan dibawahnya sehingga menjadi menjadi:  
  
\"$INCLUDE ${confdir}/sql.conf\"  
  
sqlcounter noresetcounter {  
  
counter-name = Max-All-Session-Time  
  
check-name = Max-All-Session  
  
sqlmod-inst = sql  
  
key = User-Name  
  
reset = never  
  
query = \"SELECT SUM(AcctSessionTime) FROM radacct where  
UserName=\\'%{ %k} \\\"
```

```
}
```

-- cari konfigurasi untuk accounting dan setting menjadi seperti ini dengan cara menghilangkan tanda pagarnya, selain itu berikan tanda pagar untuk yang lain –

```
accounting {
```

```
unix
```

```
radutmp
```

```
sql
```

```
}
```

-- cari juga untuk instantiate, setting dengan menghilangkan tanda pagar, seperti berikut, dan dibawah expr tambahkan

```
noresetcounter –
```

```
instantiate {
```

```
exec
```

```
expr
```

```
noresetcounter
```

```
}
```

-- cari juga pada bagian authorize, setting dengan cara yang sama, dan dibagian bawah sql tambahkan tulisan noresetcounter –

```
authorize {
```

```
preprocess
```

```
chap
```

```
mschap
```

suffix

sql

noresetcounter

}

simpan dengan ctrl+x tekan y kemudian enter

}

-- cari juga untuk instantiate, setting dengan menghilangkan tanda

pagar, seperti berikut, dan dibawah expr tambahkan

noresetcounter –

instantiate {

exec

expr

noresetcounter

}

-- cari juga pada bagian authorize, setting dengan cara yang sama,

dan dibagian bawah sql tambahkan tulisan noresetcounter –

authorize {

preprocess

chap

mschap

suffix

sql

noresetcounter

```
}
```

simpan dengan ctrl+x tekan y kemudian enter

i. Konfigurasi sertifikasi

```
#cd /etc/apache2
```

```
#mkdir ssl
```

```
#make-ssl-cert /usr/share/ssl-cert/ssleay.cnf
```

```
/etc/apache2/ssl/apache.pem
```

Contoh jawaban pertanyaan seputar SSL:

- Country Name: ID
- State Or Province Name: Jawa Timur
- Locality Name : Mergan
- Organization Name : RT?RW NET Hotspot
- Organization Unit Name : RT/RW.NET
- Host Name : localhost
- Email Address : rahadian_indr@yahoo.com

This e-mail address is being protected from spam bots, you need

JavaScript enabled to view it

Aktifkan dengan cara:

```
#a2enmod ssl
```

```
#/etc/init.d/apache2 force-reload
```

Buat dan mengenable file ssl :

```
#cd /etc/apache2/sites-available/
```

```
#nano ssl
```


-- copy konfigurasi berikut --

NameVirtualHost *:443

ServerAdmin webmaster@localhost

SSLEngine On

SSLCertificateFile /etc/apache2/ssl/apache.pem

DocumentRoot /var/www/

Options FollowSymLinks

AllowOverride None

Options Indexes FollowSymLinks MultiViews

AllowOverride None

Order allow,deny

allow from all

This directive allows us to have apache2's default start page

in /apache2-default/, but still have / go to the right place

Commented out for Ubuntu

#RedirectMatch ^/\$ /apache2-default/

ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/

AllowOverride None

Options ExecCGI -MultiViews +SymLinksIfOwnerMatch

Order allow,deny

Allow from all

ErrorLog /var/log/apache2/error.log

Possible values include: debug, info, notice, warn, error, crit,

```
# alert, emerg.
LogLevel warn
CustomLog /var/log/apache2/access.log combined
ServerSignature On
Alias /doc/ \"/usr/share/doc/"
Options Indexes MultiViews FollowSymLinks
AllowOverride None
Order deny,allow
Deny from all
Allow from 127.0.0.0/255.0.0.0 ::1/128
```

-- berakhir diatas tulisan ini –

simpan dengan ctrl+x tekan y kemudian enter

Kemudian di enable dengan cara:

```
#a2ensite ssl
```

```
#/etc/init.d/apache2 reload
```

```
#/etc/init.d/apache2 restart
```

j. Restart semua system yang dikonfigurasi

```
#/etc/init.d/apache2 restart
```

```
#/etc/init.d/mysql restart
```

```
#/etc/init.d/chilli restart
```

```
#/etc/init.d/freeradius restart
```

Atau bisa dilakukan dengan cara reboot sistem dengan mengetikkan perintah

```
#reboot
```

k. Desain web admin

Selanjutnya adalah dengan melakukan desain web admin sehingga kita tidak perlu lagi menggunakan terminal program dalam pembuatan *account*. Untuk desain web menggunakan bahasa pemrograman PHP standar sehingga mudah bagi orang awam akan bahasa pemrograman ini melakukan perubahan desain. Semua *file* yang digunakan untuk web admin disimpan dalam folder */var/www/*

4.1.4.4 *Remake with Remastersys dan Reconstructor*

Langkah terpenting dalam proses *remake* dari ubuntu ke Distro RT/RW. NET Hostpot ini adalah melakukan pembuatan *file image* ISO sehingga Distro RT/RW. NET Hostpot dapat di distribusikan kembali nantinya dengan menggunakan program *Backup Manager* setelah mendapatkan *file image* ISO dari *Backup Manager* untuk memperindah tampilan Live-CD RT/RW. NET Hostpot kita dapat menggunakan program *Reconstructor* dengan memanfaatkan *file* ISO yang kita buat tadi.

4.1.4.5 *Remastersys Backup*

Remastersys merupakan aplikasi yang sangat mudah digunakan untuk melakukan penggandaan (*Clone*) dan *data backup* distribusi linux, termasuk didalamnya *root*, *home*, *partition* dan semua informasi pribadi, *configuration* dan membuat *bootable live CD/DVD*. *Remastersys* memiliki fungsi antara lain:

1. Dapat melakukan *data backup* termasuk data pribadi kedalam Live CD atau DVD dan dapat di *instal* dan digunakan siapapun.
2. Dapat membuat CD/DVD *Distributor* dan dapat dibagikan kepada teman. Dalam CD/DVD tidak dimasukkan data pribadi.

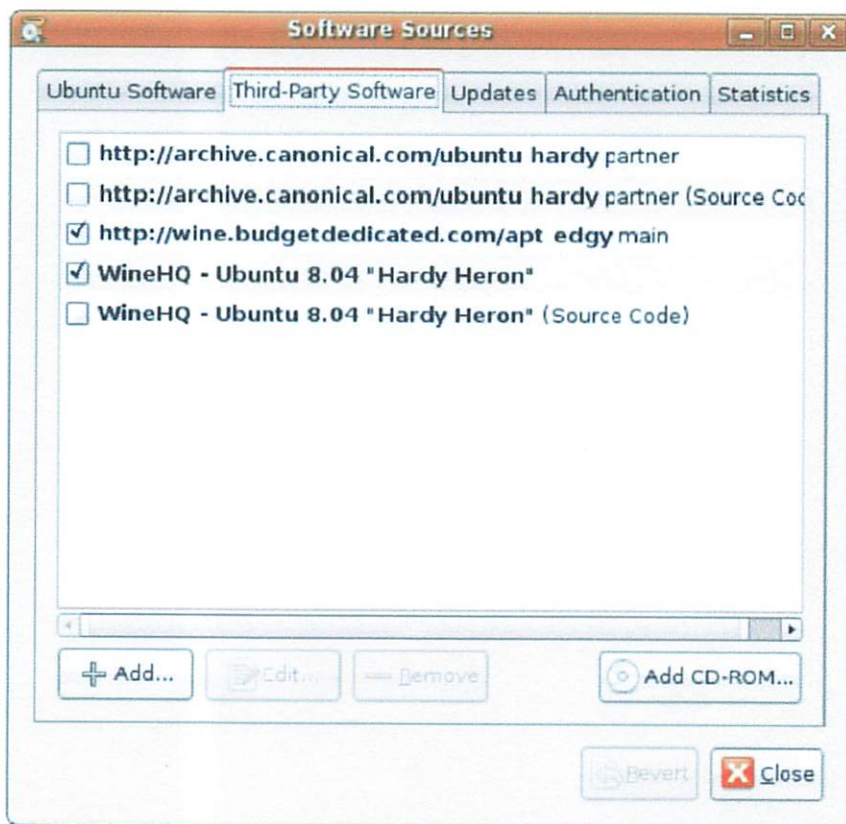
Langkah-langkah proses *data backup* menggunakan *Remastersys* adalah sebagai berikut :

a. Download *Remastersys*

Langkah yang pertama adalah mempersiapkan software *Remastersys*, file *Remastersys* dapat di *download* langsung di <http://packages.ubuntu.com>. Atau dengan memanfaatkan fasilitas yang ada ada sistem ubuntu.

Klik menu *System > Administration > Software Sources*

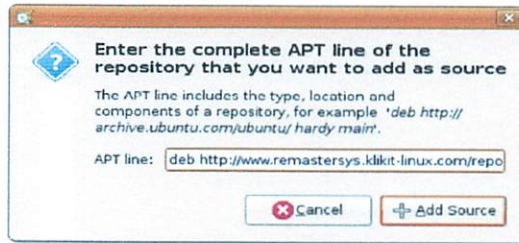
Kemudian akan muncul *toolbox Software Source* dan pilihlah *tab Third-Party Software*.



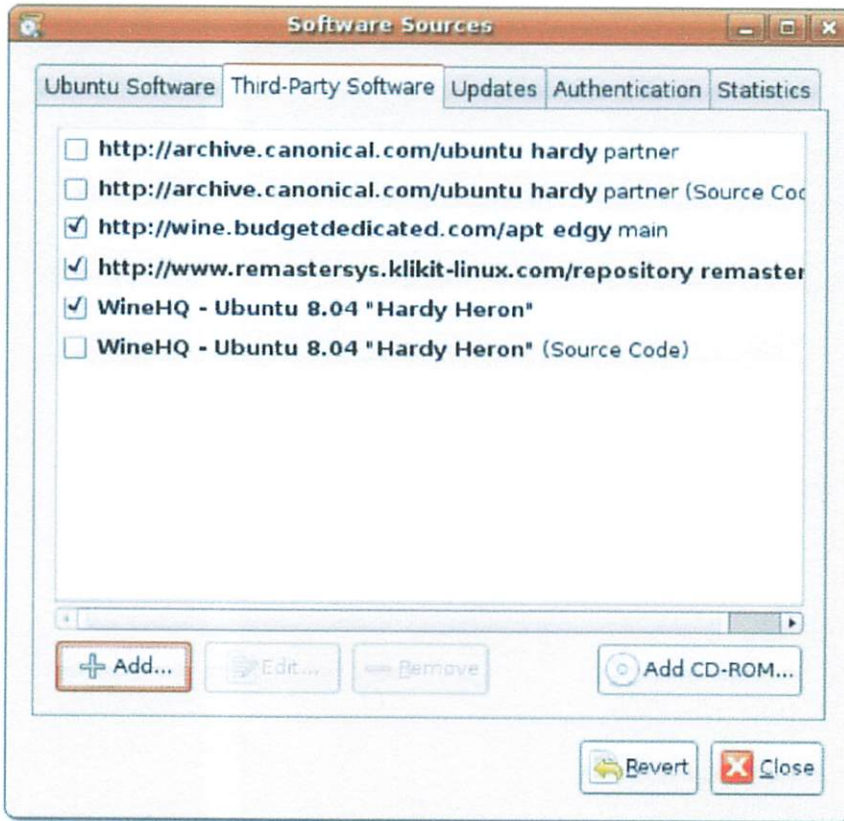
Gambar 4.1 *Jendela Software Source*

Kemudian klik tombol *Add* yang kemudian akan membuka *toolbox* baru. Masukkan alamat *Remastersys* berikut ini:

deb http://www.remastersys.klikit-linux.com/repository/remastersys/



Gambar 4.2 *Jendela address source*

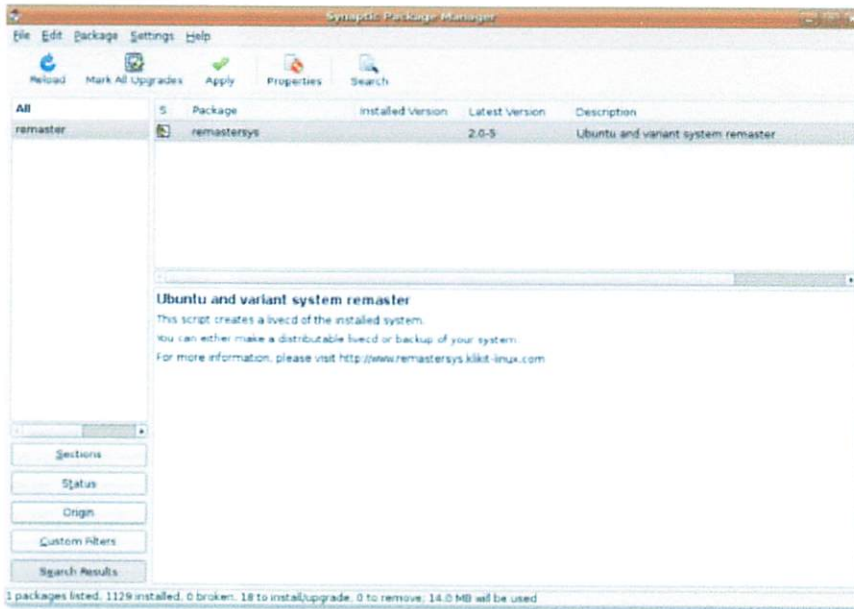


Gambar 4.3 *Jendela software source setelah dimasukkan address source yang baru*

Dan dapat anda tambahkan juga beberapa dependency *remastersys* antara lain :

- casper_1.78_i386.deb
- dmsetup_1.02.07-1ubuntu2_i386.deb
- gparted_0.2.5-1.1ubuntu11_i386.deb
- libcairomm-1.0-1_1.2.0-0ubuntu1_i386.deb
- libdebconfclient0_0.104ubuntu2_i386.deb
- libdebian-installer4_0.43_i386.deb
- libglibmm-2.4-1c2a_2.10.4-0ubuntu1_i386.deb
- libgtkmm-2.4-1c2a_1%3a2.8.8-0ubuntu1_i386.deb
- os-prober_1.14_i386.deb
- squashfs-tools_3.0-6ubuntu3_i386.deb
- ubiquity_1.2.5_i386.deb
- ubiquity-casper_1.78_all.deb
- ubiquity-frontend-gtk_1.2.5_i386.deb
- ubiquity-ubuntu-artwork_1.2.5_all.deb
- user-setup_1.2ubuntu4_all.deb

Setelah gunakan aplikasi *Synaptic Package Manager*



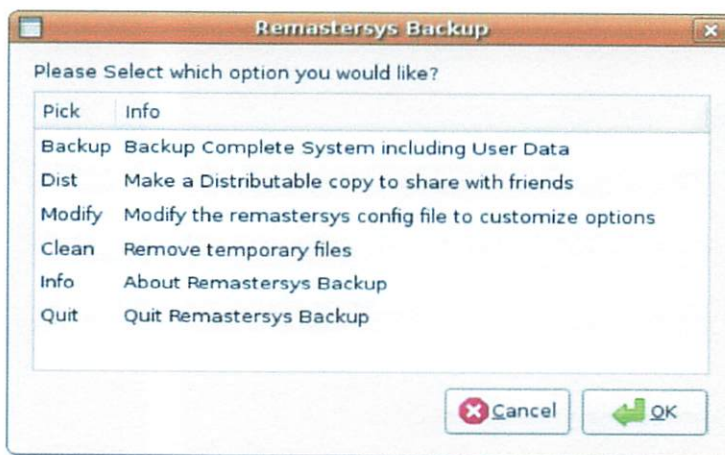
Gambar 4.4 Jendela *Synaptic Package Manager*

Pilih remastersys kemudian *install*.

b. Membuat *Distro Backup*

Aktifkan aplikasi remastersys, aplikasi ini dapat di akses pada System >

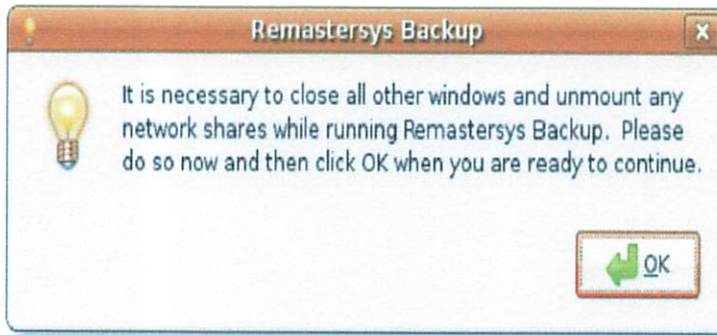
Administration > Remastersys Backup. Maka akan muncul tampilan berikut :



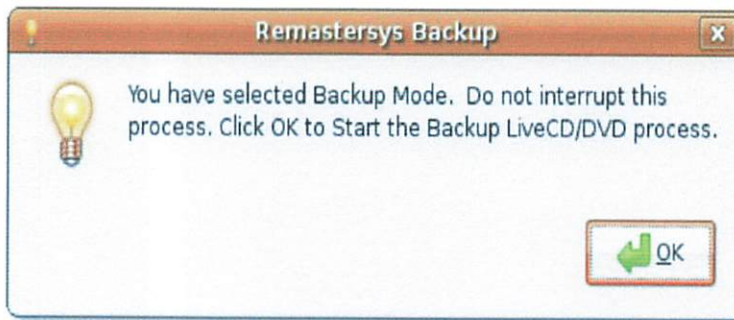
Gambar 4.5 Jendela *Remastersys*

Ada beberapa pilihan pada jendela *Remastersys* antara lain :

Backup, memungkinkan anda melakukan *backup* seluruh file system termasuk data-data *user* dan konfigurasi yang ada.

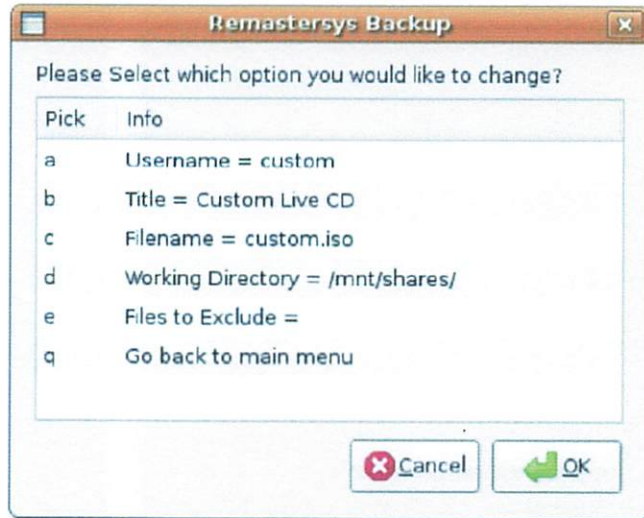


Gambar 4.6 Jendela *Remastersys Backup*



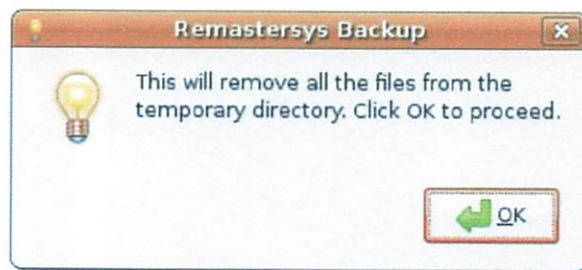
Gambar 4.7 Jendela *Remastersys Backup Confirmation*

Proses *backup* dimulai, proses ini memakan waktu beberapa menit. Hasil *image file* nantinya sekitar 750 MB, lebih baik jika dibawah itu sehingga dapat dimasukkan ke CD.



Gambar 4.10 Jendela *Remastersys Modify*

Clean, fungsinya untuk membersihkan *temporary file* yang digunakan oleh *remastersys*



Gambar 4.11 Jendela *Remastersys Clean*

Setelah memperoleh *image file*, sebaiknya dicoba terlebih dahulu menggunakan *virtual machine*.

4.1.4.6 *Reconstructor*

Sebelum menggunakan aplikasi ini mari kita mengenal apa itu *Reconstructor*. *Reconstructor* (<http://reconstructor.aperantis.com>) adalah salah satu tools untuk memudahkan kita dalam membuat atau meremaster CD Ubuntu Linux. *Reconstructor* ditulis menggunakan bahasa python dan berlisensi GNU General Public License (GPL).

CD yang dijadikan bahan remaster dengan menggunakan *Reconstructor* ini bisa menggunakan CD Desktop (*LiveCD*), *Alternate(Install)*, atau *Ubuntu Server*. Jika menggunakan basis CD Desktop (*LiveCD*) pilihan kustomisasinya sangat banyak (*add/remove software, splash, themes, fonts, wallpaper, menambah shortcut desktop dll.*)

Mengapa hanya bisa digunakan untuk Ubuntu/Xubuntu? Karena Ubuntu/Xubuntu menggunakan desktop berbasis GTK dan menggunakan Display Manager yang sama (GDM), jadi pengguna yang paling banyak mendapatkan keuntungan dari *Reconstructor* adalah anda yang me-remaster Ubuntu atau Xubuntu *LiveCD/Install CD* saja. Sedangkan anda yang menggunakan ubuntu atau lainnya, hanya mampu melakukan beberapa kustomasi saja.

Adapun langkah-langkahnya adalah sebagai berikut :

a. Instalasi *Reconstructor*

Silahkan men-*download reconstructor* di <http://reconstructor.aperantis.com>.

Saat artikel ini ditulis, tersedia reconstructor versi 2.8.

Untuk menginstall, lakukan:

```
sudo dpkg -i path/reconstructor-xx.deb
```

b. Aplikasi pendukung

Reconstructor membutuhkan aplikasi pembantu lainnya untuk memberikan hasil yang terbaik. Aplikasi tersebut antara lain :

```
python (only tested on version 2.4 and 2.5)
```

```
pygtk2
```

- | | |
|-------------------|----------------|
| 1. squashfs-tools | 8. usplash-dev |
| 2. chroot | 9. gpg |
| 3. mkisofs | 10. dpkg-dev |
| 4. gcc | 11. fakeroot |
| 5. make | 12. apt-utils |
| 6. rsync | 13. glade2 |
| 7. libbogl-dev | |

Untuk menginstal aplikasi tersebut langsung ke sistem operasi yang kita gunakan ketik perintah berikut pada terminal program :

```
sudo apt-get install squashfs-tools make gcc mkisofs rsync libbogl-dev
libusplash-dev gpg dpkg-dev fakeroot apt-utils python python-gtk2
python-glade2
```

c. Menggunakan *Reconstructor*

Untuk menjalankan *Rekonstruktör* buka folder tempat file *reconstructor* berada dan ketik perintah berikut pada terminal program :

```
sudo python reconstructor.py
```

Saat *reconstructor* sudah aktif maka *reconstructor* secara langsung akan membuka sebuah terminal program. Saat ini *reconstructor* akan mengecek keberadaan paket-paket dan *tools* serta modul yang terinstal di sistem anda.

Perhatikan pesan yang dikeluarkan *reconstructor*, pastikan anda melengkapi lebih dulu kekurangan paket sebelum memulai. Untuk melanjutkan pada jendela *reconstructor* klik tombol *next* kemudian pilih *Live CD*.

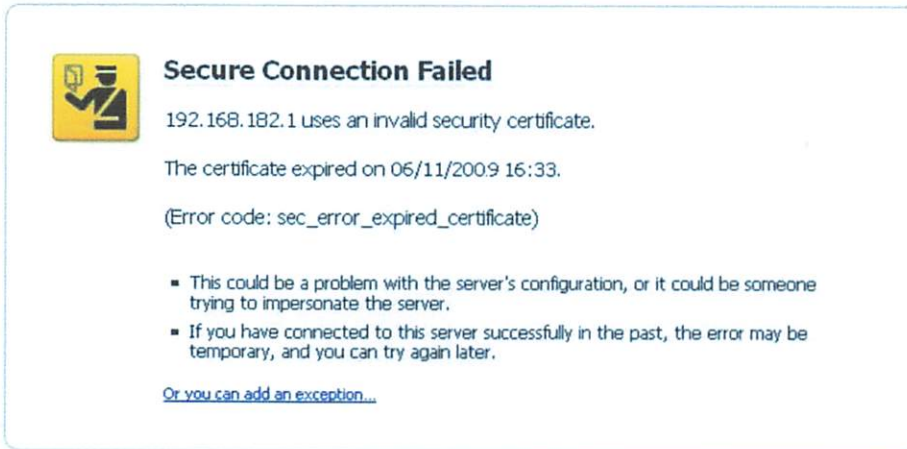
Reconstructor akan membuat beberapa direktori di home folder anda.

Secara default, *reconstructor* akan membuat sebuah direktori, *reconstructor*. dan didalamnya akan dibuat tiga direktori lagi:

- *root* untuk mengkopi isi *filesystem.squashfs* atau *live system*
- *intrad* untuk mengkompilasi *intrad* yang baru
- *remaster* untuk meletakkan file-file kopian dari CD yang akan dikompilasi menjadi ISO.

Pada saat ini kita akan menggunakan *image file* yang sudah dibuat sebelumnya dengan menggunakan *Remastersys*. Kemudian klik *Next* proses ini akan memakan waktu beberapa menit.. Saat ini anda harus menyiapkan sebuah file png dengan ukuran 640 x 400 pixel dan indexed 16 color untuk Dapper. Sedangkan untuk Gutsy anda perlu menyiapkan sebuah file png dengan ukuran 800x600 dan 256 warna indexed. Klik tombol generate untuk membuat/mengkonversi file png anda menjadi file usplash.so. Kemudian masukkan lokasinya di field usplash location. Di sini sepertinya tidak ada masalah. Anda pasti bisa melakukannya. Jika anda memiliki ikon set sendiri, letakkan di direktori `/usr/share/icons/` pada direktori kerja *reconstructor* anda (*root*). Disini kita dapat merubah ikon *start menu*. Saat ini saya tidak menggunakannya.

4.1.5 Halaman *add certificate* RT/RW NET Hotspot.

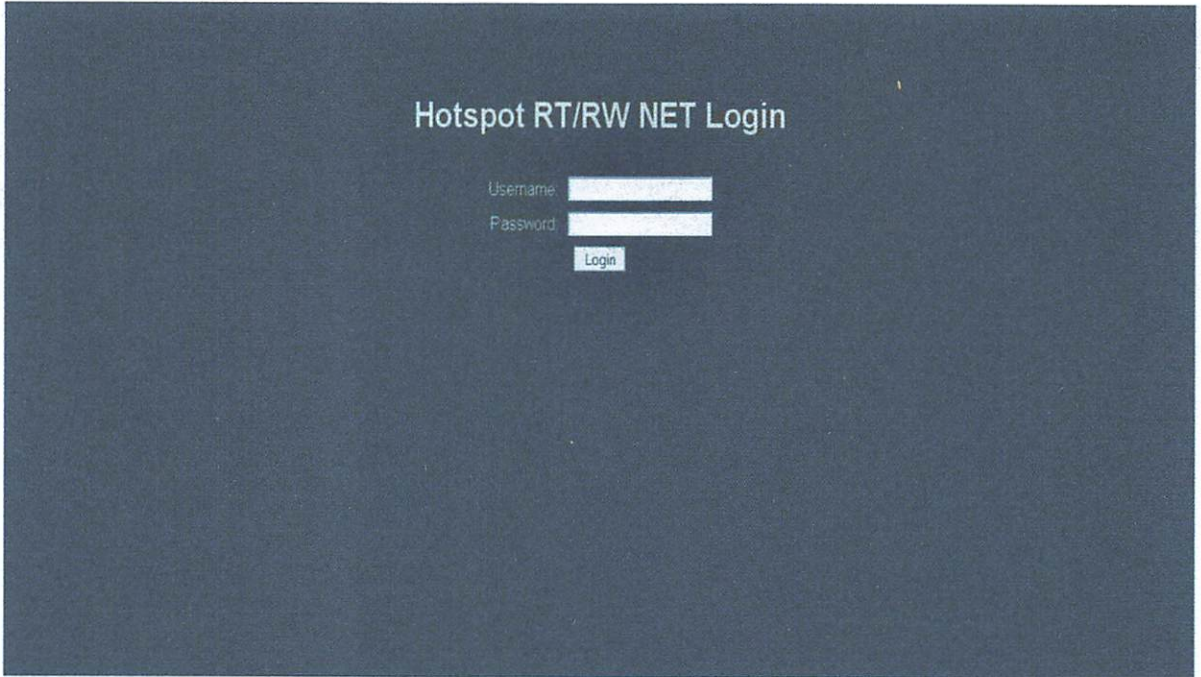


Gambar 4.12 Halaman *add certificate* RT/RW NET Hotspot dengan menggunakan *web browser* Mozilla Firefox

Bagi *user* yang akan menggunakan layanan RT/RW NET *Hotspot* akan melihat tampilan ini saat membuka *web browser* dan memasukkan alamat salah satu website yang akan anda tuju setelah terhubung ke *Acces Point*. Metode ini digunakan untuk menanamkan *encripsi file* ke dalam *computer client* yang akan digunakan dalam proses autentifikasi oleh RADIUS *server* nantinya.

4.2 Pengujian System

4.2.1 Halaman Login User

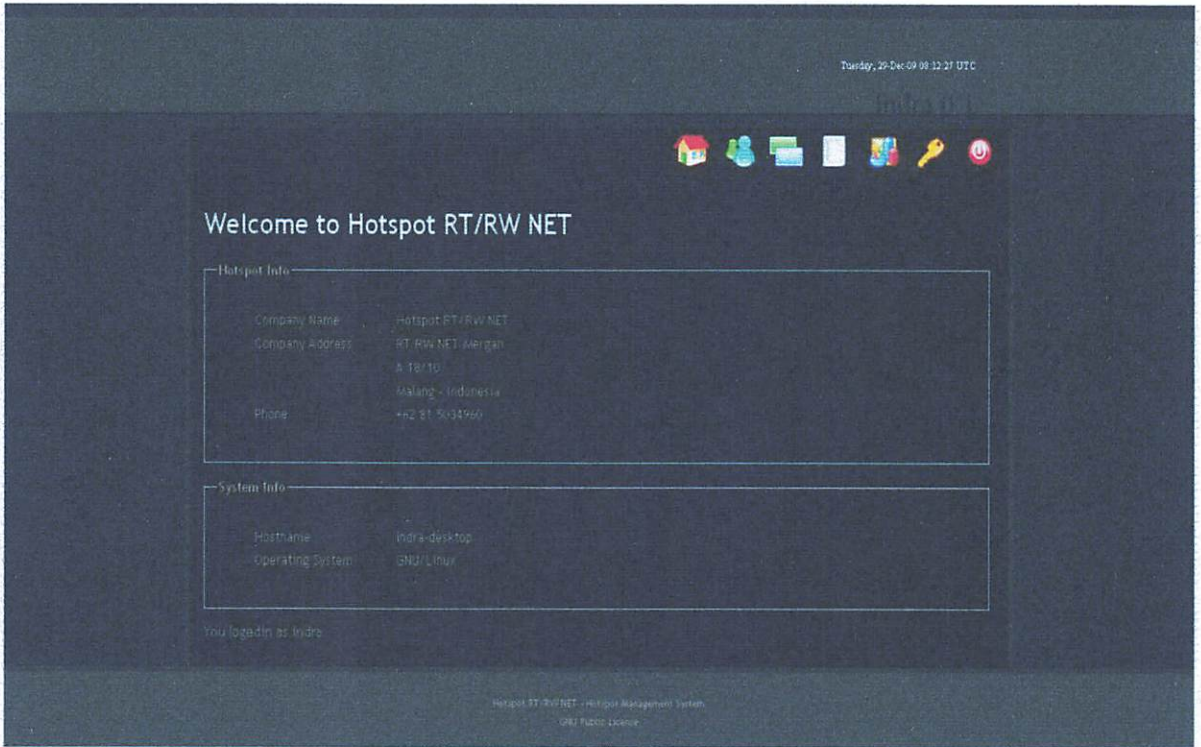


Gambar 4.13 Halaman Login User

Hanya *user* yang telah memiliki *username* dan *password* yang dapat masuk dan memperoleh dapat mengakses layanan internet. Untuk mendapatkan *username* dan *password* user yang ingin mendapatkan layanan ini dapat menghubungi langsung ke administrator *Hotspot* atau bisa melalui nomer telepon yang disediakan menyedia layanan *hostpot* ini.

Jika login berhasil maka *computer client* akan menerima *encripsi file* lagi yang akan mengalihkan *port computer client* sehingga dapat menggunakan atau mengakses internet, jika gagal maka *user* akan kembali ke halaman login, dan meminta memasukkan *username* dan *password* kembali.

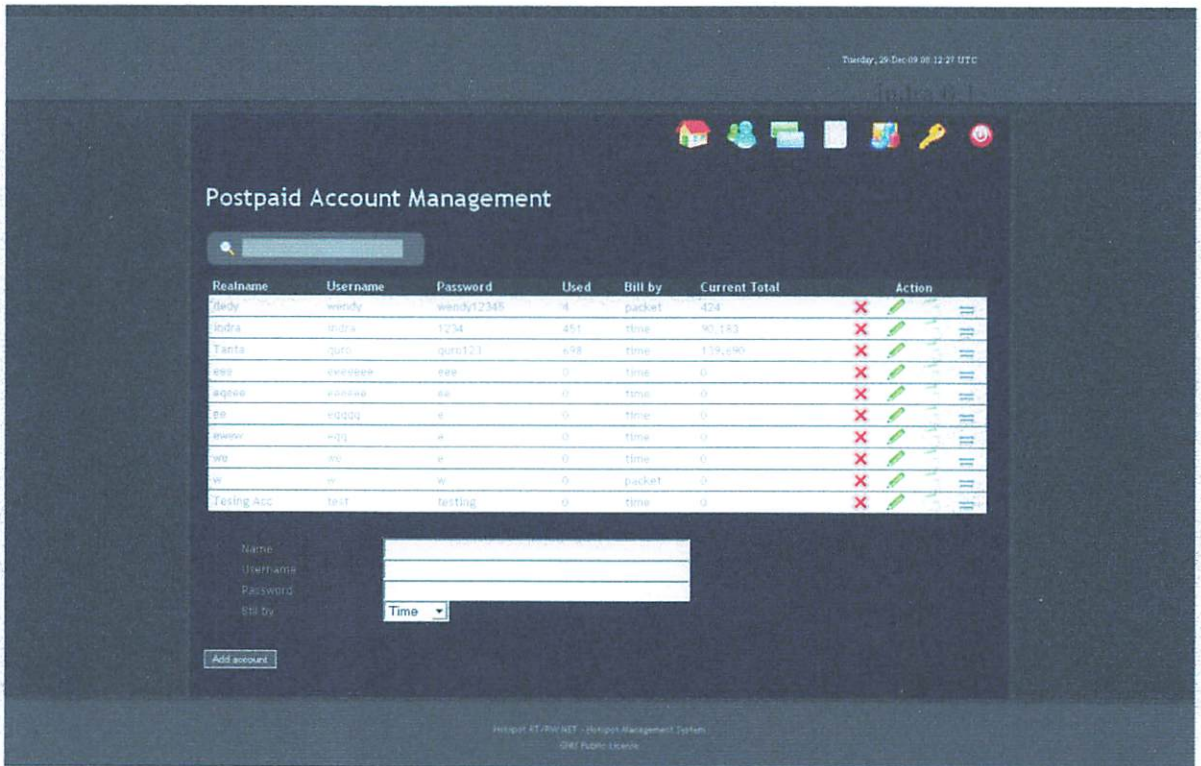
4.2.2 Halaman Home Control Panel Cashier



Gambar 4.14 Halaman Home Control Panel Cashier

Halaman *Home Control Panel Chasier* berisi informasi tentang rincian *Hotspot* yang ada mulai dari *Company Name*, *Company Address*, dan *Phone*. Serta pada halaman tersebut dilampirkan pula tipe system operasi yang digunakan oleh server *Hotspot*. Dan akhir halaman ini terdapat nama *user* yang sedang menggunakan halaman tersebut.

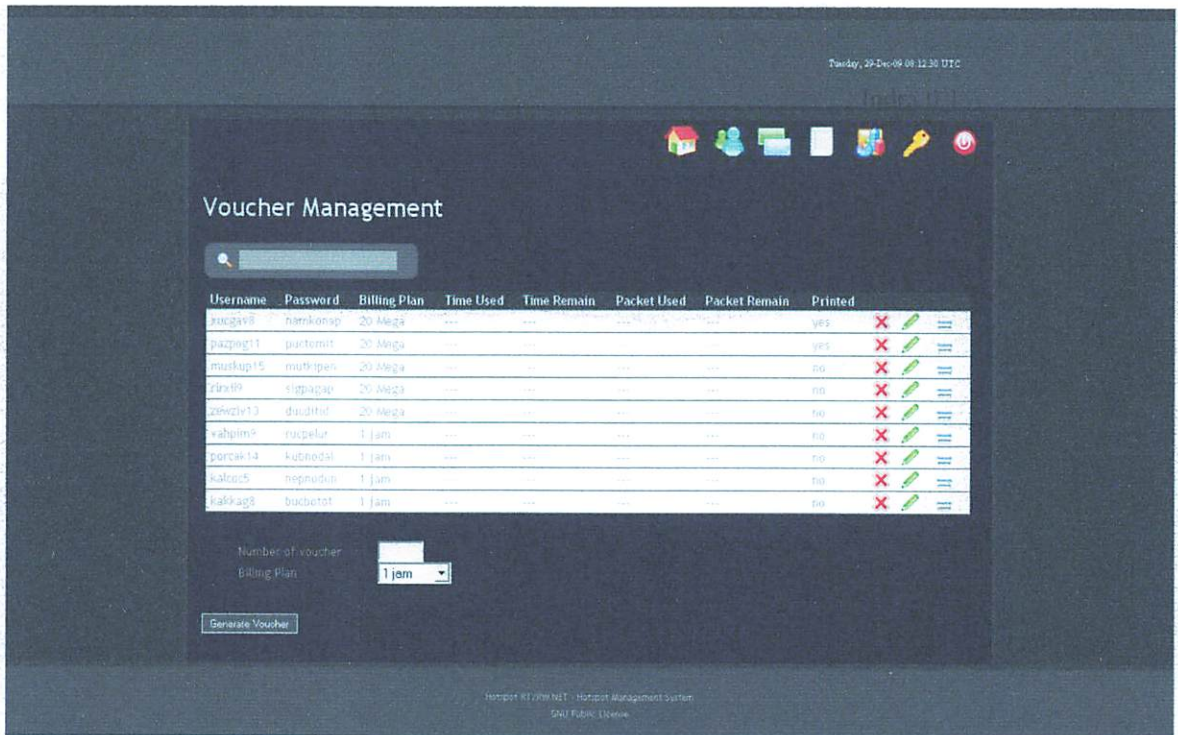
4.2.3 Halaman Postpaid Account Management



Gambar 4.15 Halaman Postpaid Account Management

Pada halaman ini kasir membuat data *user* baru atau menanbahkan *user* baru yang akan memakai fasilitas *Hostpot* RT/RW NET. Dengan memasukkan Nama, *Username*, *Password* dan *Bill By* atau tipe penggunaan *account* yang terdiri atas *Time* (waktu) dan paket (Kuota).

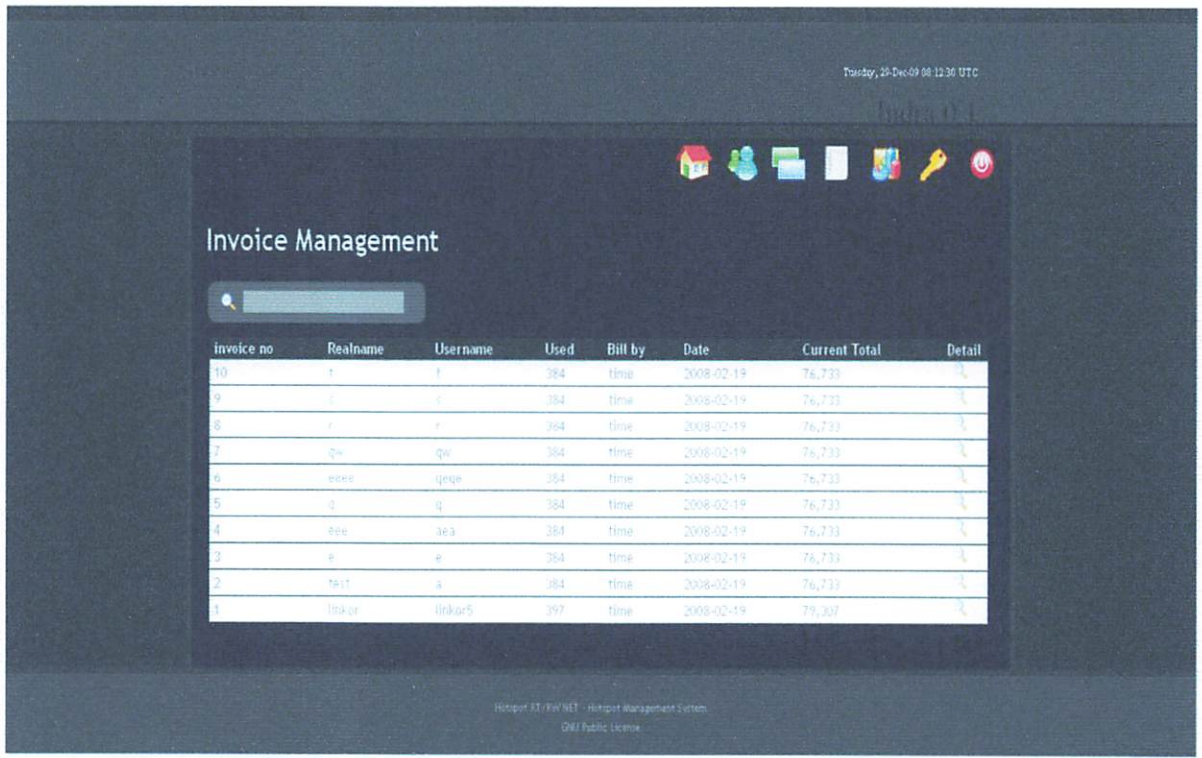
4.2.4 Halaman Voucher Management



Gambar 4.16 Halaman Voucher Mangement

Pada halaman ini kasir dapat membuat *Voucher Account* secara otomatis, dengan cara memasukkan jumlah *voucher* yang akan dibuat pada *Number of Voucher* dan memilih tipe *voucher* yang akan digunakan pada *Billing Plan*.

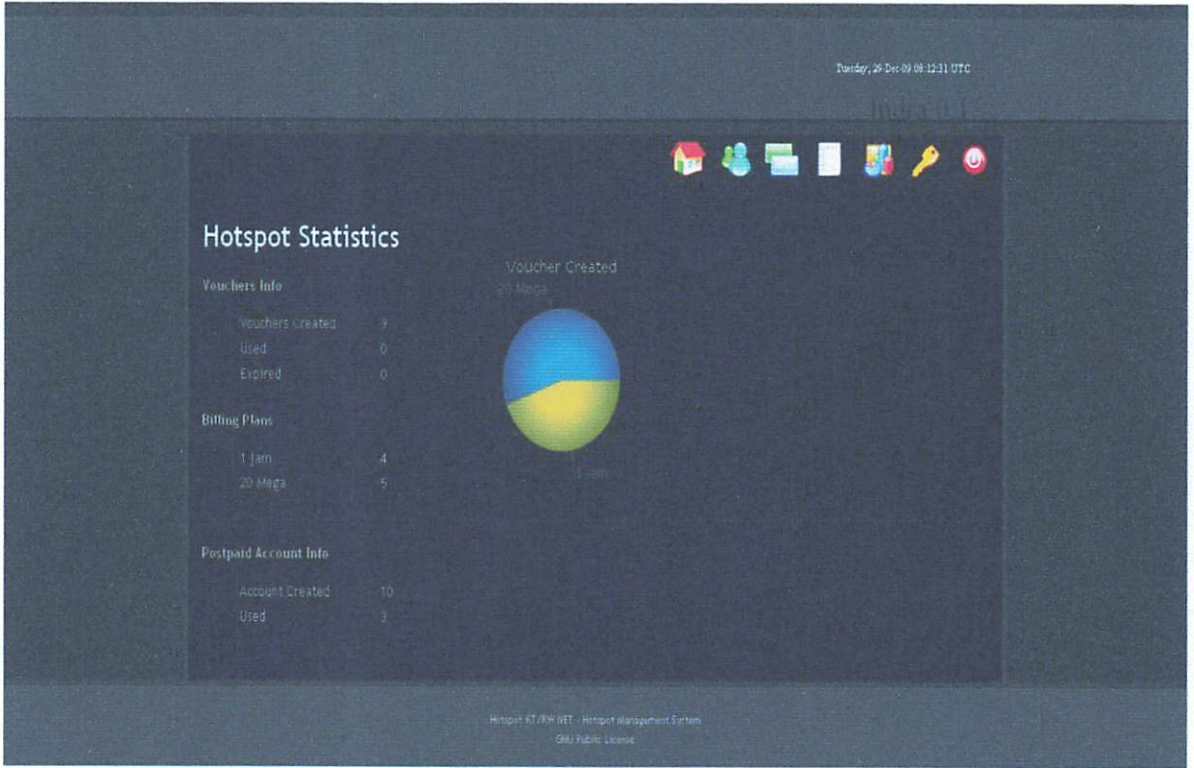
4.2.5 Halaman Invoice Management



Gambar 4.17 Halaman Invoice Management

Pada halaman ini akan diperlihatkan beberapa faktur atau yang lebih dikenal sebagai struk pemakaian *Account User* yang sudah pernah dicetak.

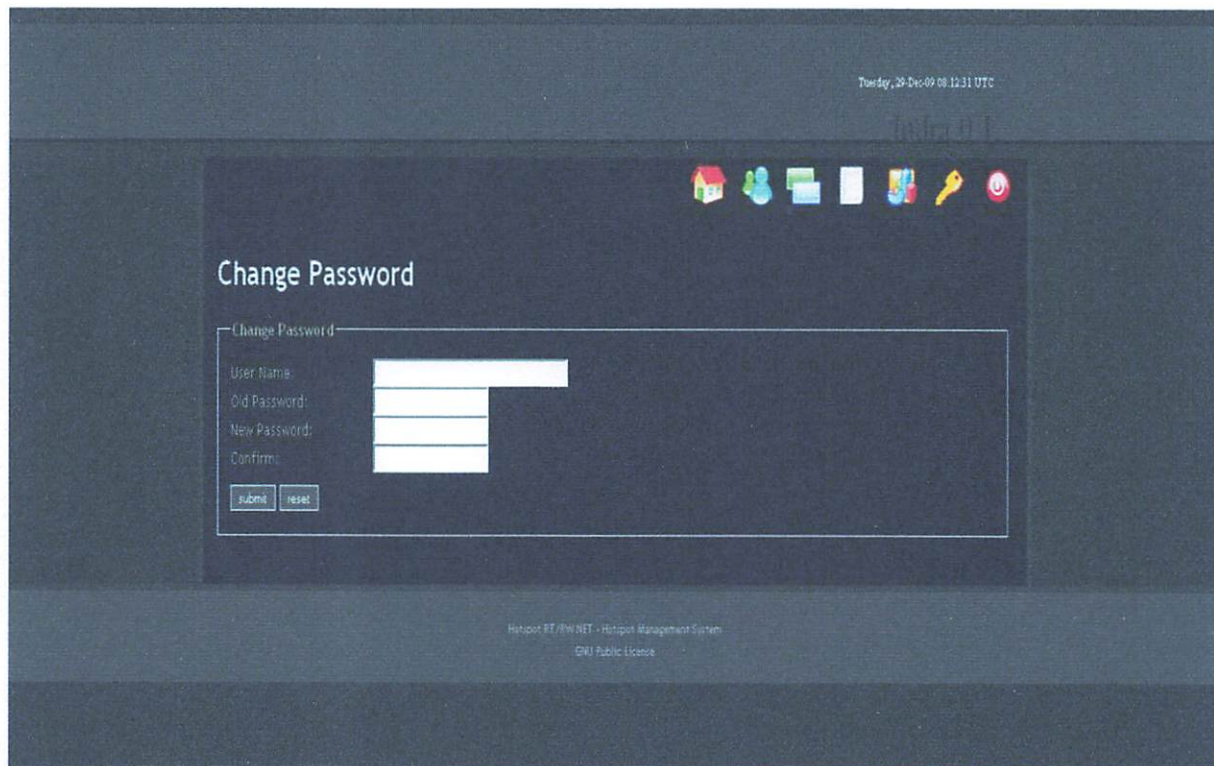
4.2.6 Halaman Statistic



Gambar 4.18 Halaman Statistic

Pada halaman ini kasir dapat mengetahui statistik pembuatan *Account User* dan pembagian *BillingPlan* yang sudah digunakan serta dapat dilihat pula pada model *Pie Chart*.

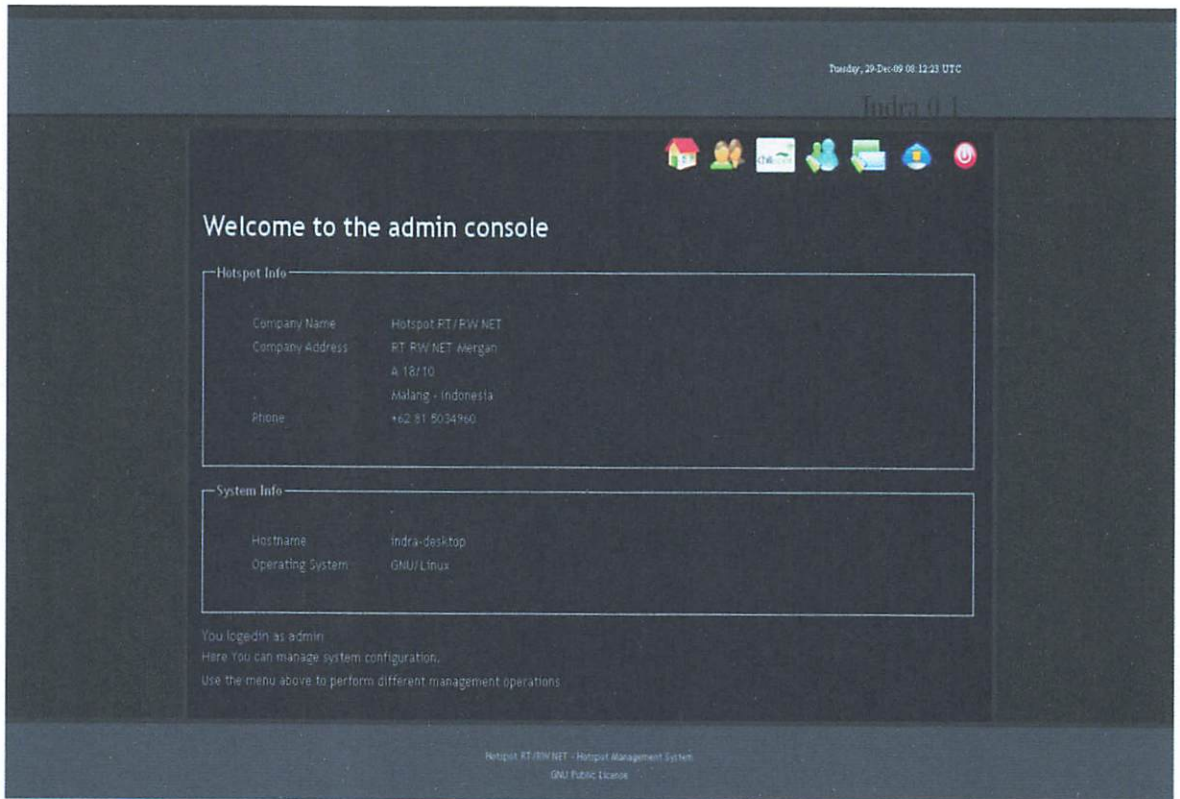
4.2.7 Halaman Chasier Change Password



Gambar 4.19 Halaman Chasier Change Password

Pada halaman ini kasir dapat melakukan perubahan *Password* hanya pada *account* yang aktif saat itu juga. Langkah ini dianggap efektif sehingga *Account Chasier* yang satu tidak dapat merubah *Password Account* yang lainnya.

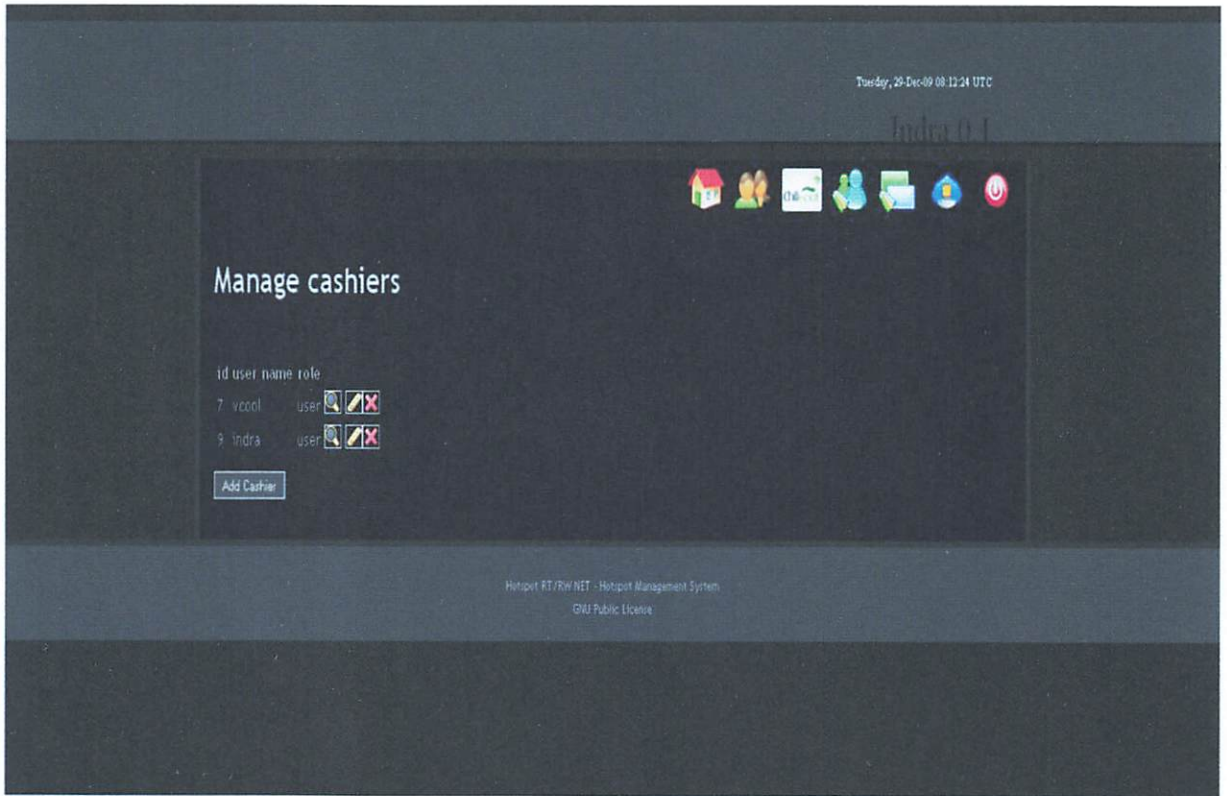
4.2.8 Halaman Home Control Panel Administrator



Gambar 4.20 Halaman *Home Control Panel Administrator*

Bila diperhatikan baik-baik tampilan *Home Control Panel Administrator* tidak jauh berbeda dengan yang dimiliki oleh *Chasier*, perbedaannya hanya pada pesan yang terdapat pada bagian bawah dari halaman ini yang menyatakan bahwa seseorang Administrator dapat mengatur konfigurasi dari sistem *Hotspot* yang digunakan.

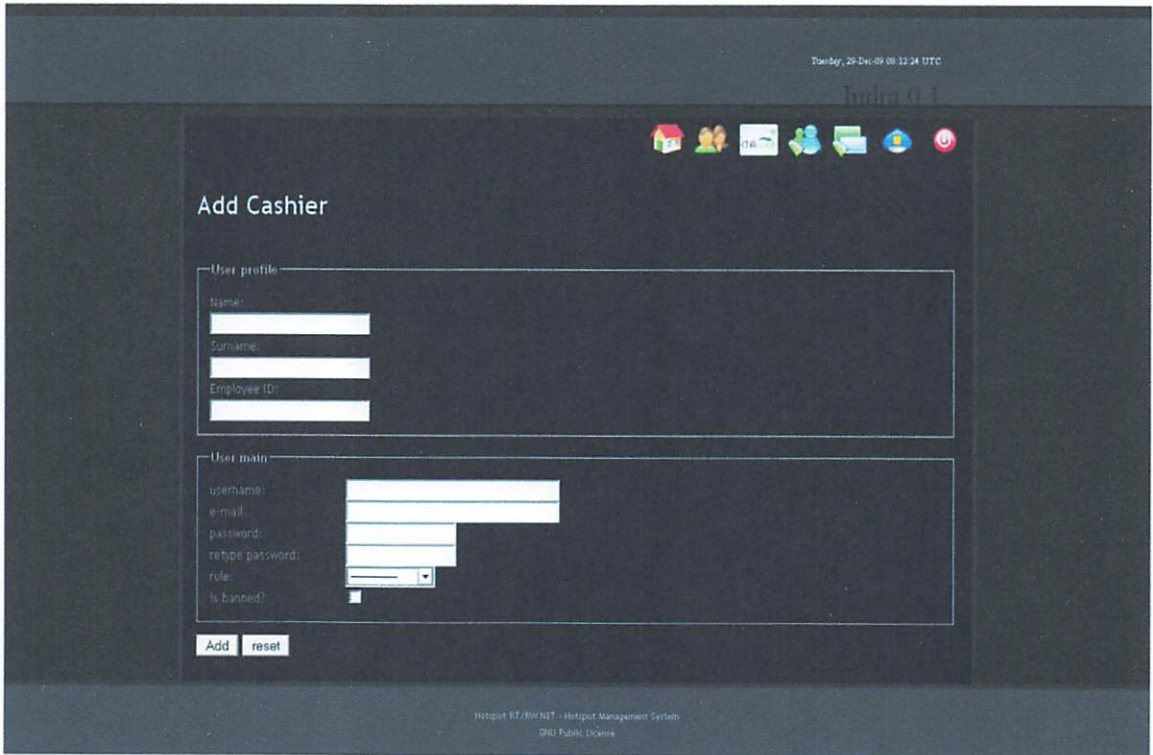
4.2.9 Halaman Cashier Management



Gambar 4.21 Halaman Cashier Management

Pada halaman ini Administrator dapat melihat daftar *Username Cashier* serta Administrator pula dapat mengedit dan menghapus *User Cashier* yang ada pada daftar tersebut.

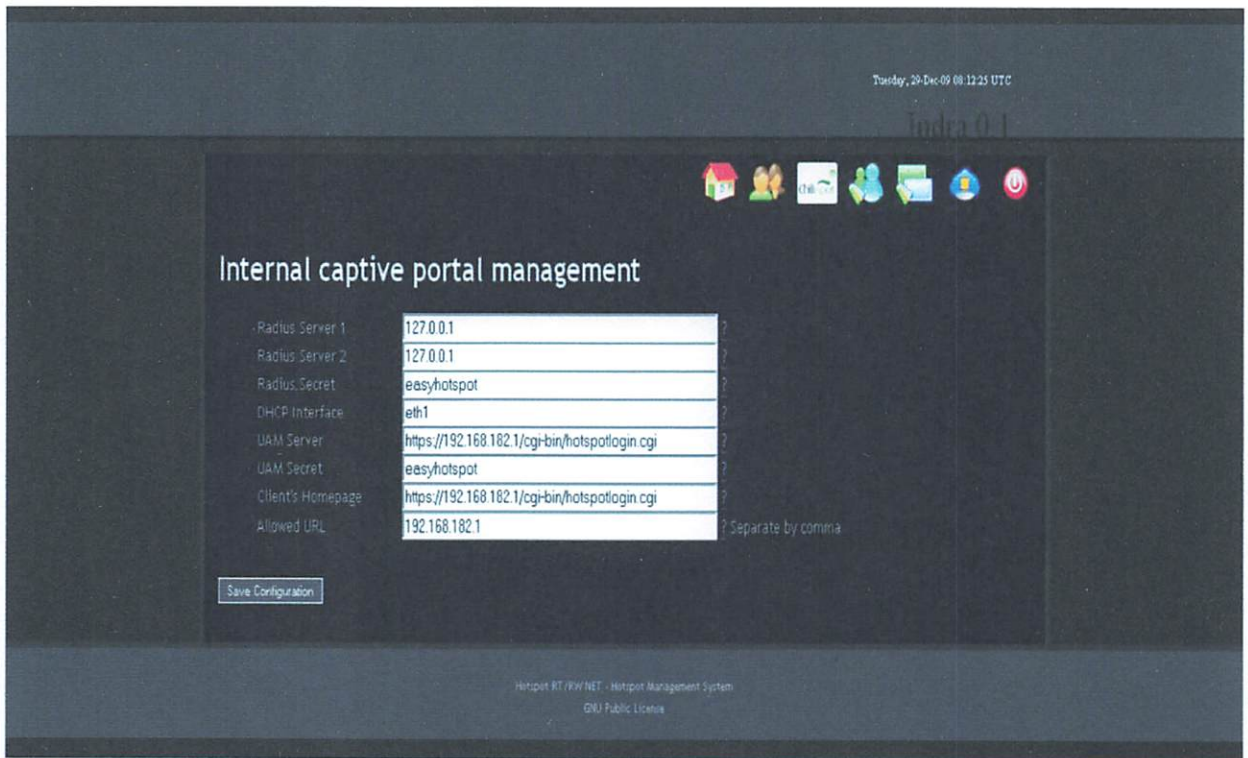
Pada halaman ini Adminstrator juga dapat membuat User Cashier yang baru. Dengan menggunakan menu *Add Chasier* yang ada sehingga akan tampak tampilan sebagai berikut :



Gambar 4 22 Halaman *Add Cashier*

Gambar diatas merupakan halaman dimana Administrator dapat membuat *account* untuk kasir yang baru dengan memasukkan beberapa data diri *Cashier* pada halaman tersebut.

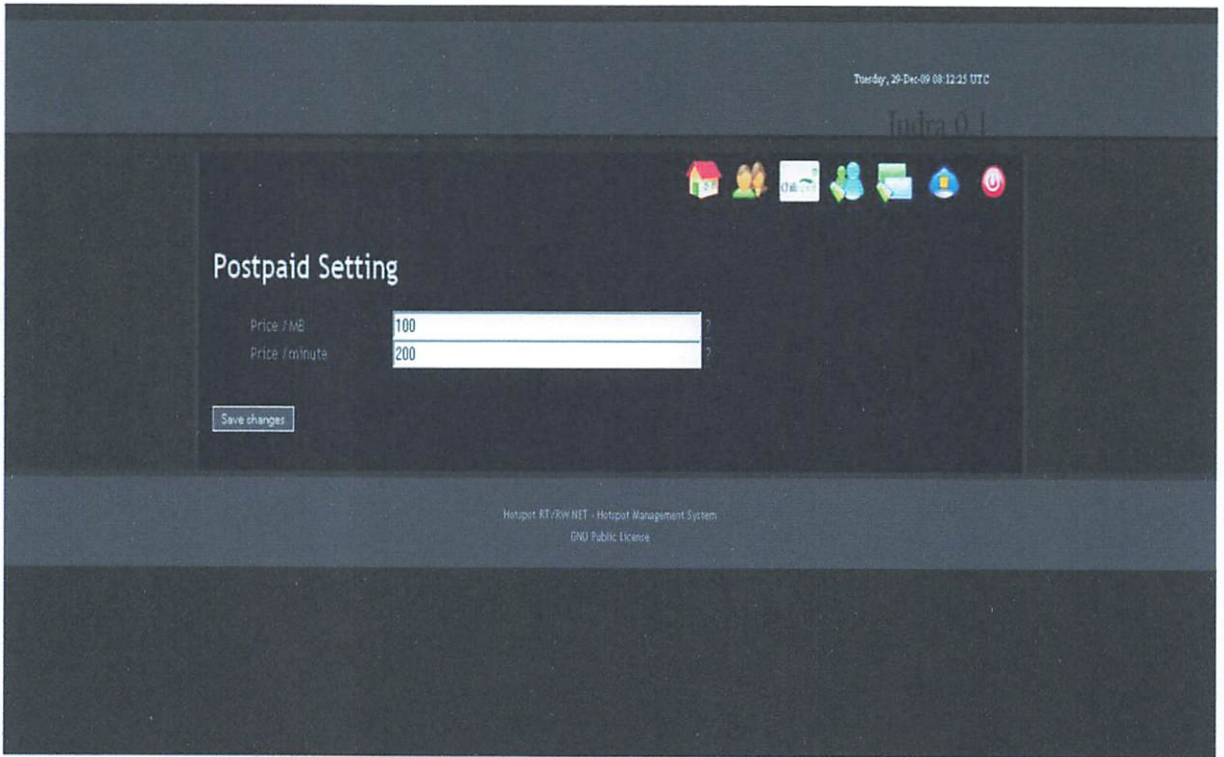
4.2.10 Halaman Chilispot Management



Gambar 4.23 Halaman *Chillispot Management*

Pada halaman ini administrator dapat mengatur settingan *Chillispot* yang ada pada *Server Hostpot Billing System* tanpa perlu aktif langsung di komputer *server*. Pada halaman ini perubahan sedikitpun pada settingannya akan berpengaruh besar pada jalannya *server*, dikarenakan setingan yang ada terhubung langsung dengan system operasi yang sedang berjalan.

4.2.11 Halaman Post Plan



Gambar 4.24 Halaman PostPlan

Pada halaman ini Administrator dapat mengatur batas limit penggunaan *Account* berdasarkan pembagian waktu dan juga pembagian berdasarkan kuota *Bandwith*.

4.2.12 Halaman Billing Plan

The screenshot displays the 'Billing Plan' management interface. At the top right, the date and time are shown as 'Tuesday, 20 Dec 09 08:12:25 UTC'. The page title is 'Indra O.I.'. Below the title, there are several navigation icons. The main content area is titled 'Billing Plan' and contains a table with the following data:

id	Name	Type	Amount	Price	DL rate	Up rate	Created by	
1	1 jam	time	60	Rp. 10,000.00	128	640	admin	✖
2	20 Mega	packet	20	Rp. 10,000.00	128	640	admin	✖

Below the table is a form to add a new billing plan. The form fields are:

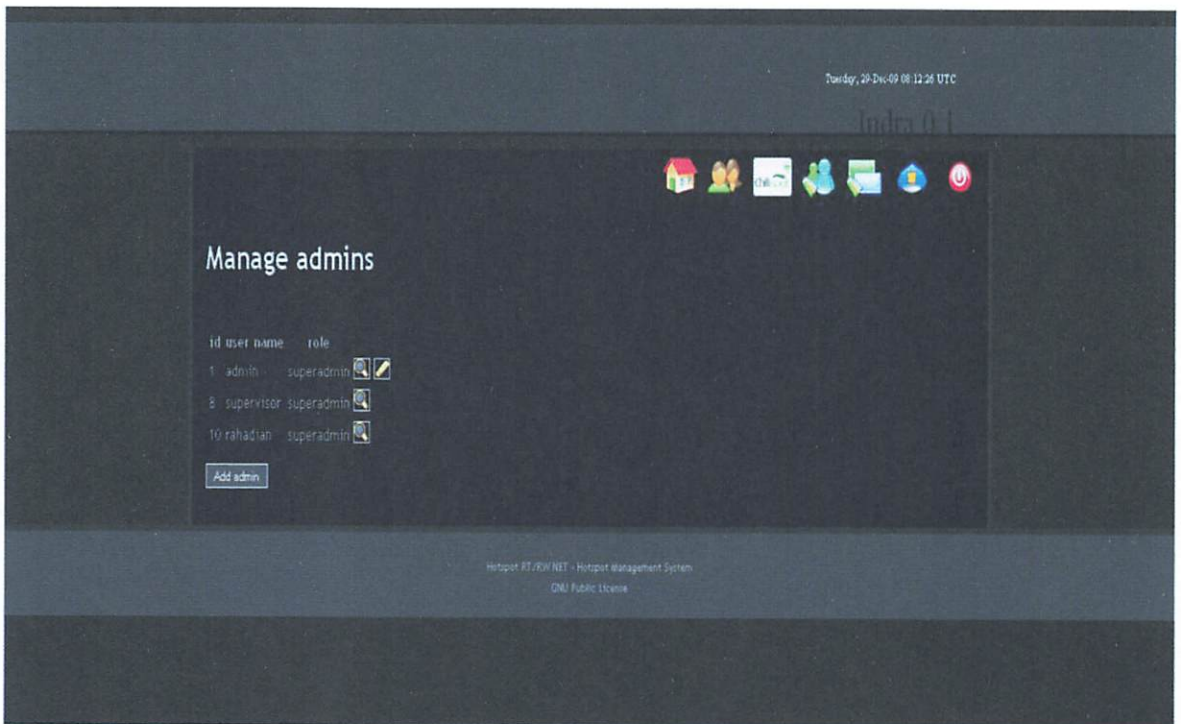
- Name:
- Type:
- Amount:
- Price:
- Download Rate:
- Upload Rate:
- Idle Timeout:

An 'Add Billing Plan' button is located at the bottom left of the form area. At the bottom of the page, the footer text reads: 'Hotspot RT/PW/NET - Hotspot Management System' and 'GAG Public License'.

Gambar 4.25 Halaman *Billing Plan*

Pada halaman ini Administrator dapat membagi *Post Plan* menjadi beberapa bagian lagi sehingga memudahkan *Cashier* dalam pembuatan *Account User* nantinya, yakni dibagi berdasarkan tipe pengguna serta batas-batas limit pemakaian berdasarkan waktu dan juga kuota yang akan digunakan oleh *user*.

4.2.13 Halaman Security



Gambar 4.26 Halaman Security

Pada halaman ini akan di tampilkan daftar *username Administrator* yang ada. Dan juga Administrator dapat menambah *Account* yang baru dengan menggunakan tombol Add Admin.

Bab V

PENUTUP

5.1 Kesimpulan.

Berdasarkan pembahasan di atas dapat ditarik beberapa kesimpulan, yaitu sebagai berikut :

1. *Hotspot Billing System* ini dirancang sehingga dapat mengatasi permasalahan manajemen yang sering terjadi dan dapat digunakan dengan mudah oleh operator jaringan yang ada di wilayah RT/RW.
2. *Hotspot Billing System* ini merupakan *billing system* yang berbasis *Open Source*, dimana *billing* ini cara menginstalnya atau pun cara pengoperasiannya relatif mudah. Serta dapat di modifikasi dan perlu diketahui aplikasi *Open Source* ini bersifat *free*.
3. Setelah di uji coba aplikasi ini dapat berjalan pada beberapa macam OS, seperti : XP, vista, windows 7.

5.2 Saran.

Berdasarkan kesimpulan hasil penelitian di atas, maka dapat diajukan beberapa saran, yaitu sebagai berikut :

1. Perlu adanya sebuah media yang dapat mengontrol dan mengelola jaringan *Wireless* yang ada di suatu kalangan RT/RW karena dengan menggunakan settingan manual mengakibatkan membutuhkan perhatian khusus dari operator jaringan.
2. Media Manajemen jaringan *Wireless* atau *Hotspot Billing System* ini tidak hanya dapat di implementasikan di kalangan masyarakat saja, melainkan

dapat pula dimanfaatkan oleh semua kalangan . seperti di kalangan pendidikan.

3. Apabila user dalam pemakaian nya melebihi kuota yang di tentukan.. perlunya pengembangan dalam mengatasi hal ini.

Daftar Pustaka

- [1] C. Lucas Jr, Henry. *Analisis, Desain, Dan Implementasi Sistem Informasi*. Erlangga, Jakarta. 1993.
- [2] <http://digda.blog.uns.ac.id/2009/06/09/konsep-kerja-php/>
- [3] Kurniawan, Yahya. *Aplikasi Web Database dengan PHP dan MySQL*. PT. Elex Media Komputindo, Jakarta. 2002
- [4] <http://en.wikipedia.org/wiki/RADIUS>
- [5] <http://www.scribd.com/doc/19376884/Autentifikasi-Radius>
- [6] <http://vikar-permana.blogspot.com/2009/03/chillispot.html>
- [7] <http://en.wikipedia.org/wiki/Internet>
- [8] <http://id.wikipedia.org/wiki/Hotspot>
- [9] www.wikipediaindonesia.com
- [10] www.ilmukomputer.com
- [11] <http://www.ubuntu.com/>

LAMPIRAN



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK KOMPUTER & INFORMATIKA S-1
Jl. Karanglo KM. 2 Malang

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama : RAHADIAN INDRA NUGROHO
NIM : 04 12 616
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer & Informatika
Judul Skripsi : Pengembangan Aplikasi *Billing System* Berbasis *Open Source* di RT/RW NET

Dipertahankan di hadapan majelis penguji Skripsi jenjang Strata satu (S-1) pada :

Hari : Senin
Tanggal : 8 Februari 2010
Dengan Nilai : 75,8 (B+) *BY*



Ir.H.Sidik Noetjahjono, MT
NIP.Y. 1028700163

Sekretaris Majelis Penguji

Ir. F. Yudi Limpraptono, MT
NIP.Y. 1039500274

Dosen Penguji I

Komang Somawirata ST, MT
NIP.P.1030100361

Dosen Penguji II

Irmalia Suryani Faradisa, ST, MT.
NIP. P.1030000365



FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer & Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Rahadian Indra Nugroho
NIM : 04 12 616
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer & Informatika
Masa Bimbingan : 25 Desember 2009 s/d 25 Juni 2010
Judul Skripsi : Pengembangan Aplikasi *Billing System* Berbasis *Open Source* di RT/RW NET

Penguji/ Tanggal	Uraian	Paraf
Penguji I	1. Kesimpulan di bandingkan dengan yang sudah ada	
	2. Demo untuk portal	
Penguji II	3. Flowchart	
	4. Kesimpulan	
	5. Abstrak	

Dosen Penguji,

Penguji I

I Komang Somawirata ST, MT
NIP:P.1030100361

Penguji II

Irmalia Suryani Faradisa, ST., MT.
NIP. P.1030000365

Mengetahui,

Dosen Pembimbing I

Ir.Th.Mimien Mustikawati MT
NIP. P. 1030000352

Dosen Pembimbing II

Sot'yohadi ST.
NIP.Y.1039700309



FORMULIR BIMBINGAN SKRIPSI

Nama : RAHADIAN INDRA NUGROHO
 Nim : 04.12.616
 Masa Bimbingan : 25 DESEMBER 2009 s/d 25 JUNI 2010
 Judul Skripsi : PENGEMBANGAN APLIKASI BILLING SYSTEM BERBASIS
 OPEN SOURCE di RT/RW NET

No	Tanggal	Uraian	Paraf Pembimbing
1	11-01-2010	Bcb I - IV revisi	
2	13-01-2010	Bcb I - IV uk	
3	16-01-2010	Seminar hasil	
4	02-02-2010	Pemo Aplikasi	
5	03-02-2010	Bcb V + Lampiran	
6	05-01-2010	Kumpro	
7			
8			
9			
10			

Malang,

Dosen pembimbing I

Ir. TH. Mimien Mustikawati, MT

NIP.P.1030000352



INSTITUT TEKNOLOGI NASIONAL
 Jl. Bendungan Sigura – gura No.2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : **RAHADIAN INDRA NUGROHO**
 Nim : **04.12.616**
 Masa Bimbingan : **25 DESEMBER 2009 s/d 25 JUNI 2010**
 Judul Skripsi : **PENGEMBANGAN APLIKASI BILLING SYSTEM BERBASIS
 OPEN SOURCE di RT/RW NET**

No	Tanggal	Uraian	Paraf Pembimbing
1	12-01-2010	Bab I & Bab II revisi	<i>fadi</i>
2	14-01-2010	Bab I & Bab II Acc	<i>fadi</i>
3	18-01-2010	Bab III - Bab V revisi	<i>fadi</i>
4	21-01-2010	Bab III - Bab V Acc	<i>fadi</i>
5	26-01-2010	Makalah Seminar hasil	<i>fadi</i>
6	05-02-2010	Demo Aplikasi	<i>fadi</i>
7			
8			
9			
10			

Malang,

Dosen pembimbing II

Sotyo Hadi ST.
NIP.Y.103970309

Form S-4b

Jan.



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

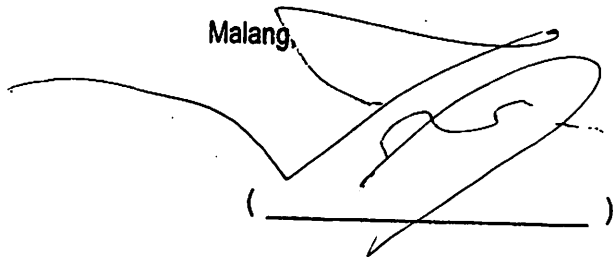
Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : RAHMADHAN INDRA N.
NIM : 201.12.616.
Perbaikan meliputi :

- * Kesimpulan. Pibandingkan dgn yg udh ada !
- * Dams uthe portal !

Malang,





INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : RAHARDIAN INDRAN
NIM : 24.12.666
Perbaikan meliputi :

- * Kesimpulan. Di landingkan dgn yg udh ada!
- * Dams uthe portal!

Malang,