

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini dijelaskan mengenai implementasi hasil rancang bangun sistem, penggunaan program aplikasi yang dibuat, serta pengujian terhadap sistem. Bab ini terdiri dari tiga sub bab, yaitu: Sub Bab 4.1 Implementasi Sistem berisi penjelasan mengenai kebutuhan hardware dan software untuk server dan client serta penjelasan tahapan instalasi program; Sub Bab 4.2 Implementasi program berisi penjelasan dari semua aplikasi pendukung yang dibuat, baik dari sisi eksternal, internal dan pihak lain yang terkait; dan yang terakhir adalah Sub Bab 4.3 Evaluasi berisi hasil pengujian terhadap semua fasilitas dan fungsi yang disediakan oleh sistem.

4.1 Implementasi Sistem

4.1.1 Kebutuhan hardware / software

Perangkat lunak dan perangkat keras yang digunakan dalam pembuatan sistem ini terbagi menjadi dua, yaitu :

1. Server

Software yang diperlukan untuk dapat menjalankan aplikasi ini adalah :

- a. OS Redhat Linux 9
- b. FreeRADIUS 1.0
- c. Apache versi 1.0
- d. Internet Explorer 5.0
- e. MySQL versi 3.23
- f. PHP versi 4.3.4

Sedangkan hardware yang dibutuhkan sebaiknya :

- a. Server menggunakan mesin HP Proliant DL 380 G-3
- b. Pentium 4, 306 GHz
- c. Memory SDRAM 1 Gb
- d. Harddisk 3 x 36 Gb
- e. Minimal VGA Card 32 MB
- f. Monitor dengan resolusi minimal 800 x 600 pixel

2. Client

Software yang diperlukan untuk dapat menjalankan aplikasi ini adalah :

- a. Windows 98 SE
- b. Internet Explorer 5.0

Sedangkan hardware yang dibutuhkan :

- a. Minimal komputer dengan prosessor Pentium 233 MHz
- b. Memory SDRAM 256 MHz
- c. Harddisk 4,3 Gigabyte
- d. Minimal VGA Card 4 MB.
- e. Monitor dengan resolusi minimal 800 x 600 pixel
- f. Koneksi Internet

4.1.2 Instalasi program

Untuk menginstal program aplikasi TELKOMNet@Home berbasis Internet dan Intranet ini dapat dilakukan dengan tahapan sebagai berikut :

- a. Pastikan Apache web server telah terinstall di PC anda, apabila belum silahkan install dengan menggunakan installer Apache HTTP Server for windows versi 1.3.22.

b. Lakukan instalasi PHP dengan menjalankan installer PHP versi 4.3.4 for windows. Ada dua macam cara penginstalan php, yaitu instalasi php sebagai CGI dan sebagai modul ISAPI. Cara kedua lebih direkomendasikan dikarenakan sesuai dengan referensi yang ada memiliki kelebihan dalam hal kecepatan pemrosesan halaman php yang lebih baik. Adapun cara instalasi php sebagai modul ISAPI dapat dijelaskan secara singkat sebagai berikut:

1. Jalankan file installer php 4.3.4 dengan direktori instalasi standar pada **“C:\PHP”**.
2. Rename file **“C:\PHP\php.ini-recommended”** menjadi **“C:\PHP\php.ini”**, lalu edit dengan menggunakan notepad.
3. Ubah baris pada kalimat **;variables_order = "EGPCS"** menjadi **variables_order = "EGPCS"** dengan menghilangkan tanda **;**. Langkah ini diperlukan untuk memungkinkan php menerima request data dari klien baik dalam bentuk POST atau GET request.
4. Ubah baris kalimat **;session.save_path** dengan menghilangkan tanda **#** didepannya lalu isikan folder tempat php akan menyimpan informasi cookies dari tiap pengguna yang melakukan akses web. Misal: **session.save_path= “C:\Program Files\Apache Group\Apache\session”**.
5. Aktifkan modul XSLT dengan cara menghilangkan tanda **;** pada baris **;extension=php_xslt.dll**. Langkah ini diperlukan agar halaman php yang telah dibuat dapat memanfaatkan fungsi-fungsi XSLT.

6. Aktifkan modul GD dengan cara menghilangkan tanda ; pada baris **`;extension=extension=php_gd2.dll`**. Langkah ini diperlukan agar halaman php yang telah dibuat dapat memanfaatkan fungsi-fungsi grafik.
7. Langkah terakhir adalah melakukan perubahan file konfigurasi apache yang terletak di **“C:\Program Files\Apache Group\Apache\conf\httpd.conf”** dengan menambahkan baris berikut:

```
LoadModule php4_module c:/php/sapi/php4apache.dll
```

```
AddModule mod_php4.c
```

```
AddType application/x-httpd-php .php
```

- c. Copykan source program pada direktori root yang secara default terletak pada **“C:\Program Files\Apache Group\Apache\htdocs”**.
- d. Restart Apache web server dengan menjalankan shortcut yang terletak pada **Start>Program>Apache HTTP Server>Control Apache Server>Restart**.

Untuk menjalankan program, buka Internet Explorer, lalu pada address bar ketikkan **http://localhost/home/index.php** untuk membuka sistem eksternal bagi para pelanggan, serta **http://localhost/home/admin/index.php** untuk membuka aplikasi sistem internal.

4.2 Implementasi program

Secara garis besar, sistem pendukung layanan TELKOMNet@Home terdiri dari aplikasi internal, aplikasi eksternal, program penjadwalan dan program console update data pelanggan, ditambah dengan dua buah aplikasi tambahan sebagai contoh studi kasus pemanfaatan teknologi XML, yaitu program simulasi rekening telepon (Intagjastel) dan print out rincian penggunaan TELKOMNet@Home. Dengan demikian pembahasan implementasi sistem akan

dibagi menjadi lima sub pembahasan yang masing-masing akan menjelaskan secara umum dari detail tiap-tiap sistem.

4.2.1 Sistem aplikasi eksternal

Sistem aplikasi eksternal dibuat agar para pengguna jasa layanan TELKOMNet@Home dapat memperoleh informasi tagihan beserta detail penggunaan pada setiap bulannya, dan fasilitas penggantian password akses layanan TELKOMNet@Home serta seting kontrol waktu akses. Dimana pelanggan yang ingin memanfaatkan fasilitas tersebut dapat mengunjungi alamat situs *www.telkomnet@home.com*. Berikut adalah tampilan depan web aplikasi eksternal layanan TELKOMNet@Home:



Gambar 4.1 Tampilan depan aplikasi web eksternal.

a. Sistem autentifikasi pengguna web eksternal

Sebelum pelanggan layanan TELKOMNet@Home dapat menggunakan fasilitas yang tersedia pada web layanan eksternal, terlebih dahulu melakukan proses login dengan memasukkan username dan password, untuk kemudian sistem akan melakukan pencocokan dengan data yang tersedia di **tabel pelanggan** pada **database AtHome2** . Berikut adalah potongan program untuk melakukan pencocokan username-password:

Program 4.1 Program pencocokan username-password dengan data pada tabel pelanggan

```

<?php
include_once "connect.php";
session_start();

function password_valid($cust_user,$cust_pass) {
    $login_name="root";
    $password="";
    $host="Ahome2";
    $con=dbconnect($login_name,$password,$host);
    $query="SELECT * FROM pelanggan WHERE user_home='$cust_user' and
status_aktifasi='1' ";
    dbquery ($con,$query,$stmt,$d);
    if ($rows=dbfetch($stmt)){
        $pass_valid=dbresult($rows,"pass_home");
        $_SESSION['nama_user']=dbresult($rows,"nama");
        if($pass_valid==$cust_pass) {
            return TRUE;
        }else{
            return FALSE;
        }
    }else{
        return FALSE;
    }
}
?>

```

b. Sistem informasi profil pelanggan

Setelah pelanggan sukses melakukan proses login, halaman pertama yang akan ditampilkan adalah halaman profil pelanggan. Halaman ini dibuat dengan menggunakan php yang akan mentransformasi data XML profile pelanggan (file C:\Program Files\Apache Group\Apache\htdocs \home\cust\cust.xml) dengan file

XSLT (file C:\Program Files\Apache Group\Apache\htdocs\home\cust\pars_xsl_plg.xml) untuk kemudian dihasilkan halaman HTML profile pelanggan (<http://localhost/home/cust/index.php>).

Program 4.2 Program transformasi file cust.xml dengan file pars_xsl_plg.xml

```

$xml_file = "cust.xml";
$xml_file = "pars_xsl_plg.xml";

$vars = explode("&", $_SERVER['QUERY_STRING']);
for ($i=0;$i<=count($vars);$i++) {
    $var = explode("=", $vars[$i]);
    $params[$var[0]] = $var[1];
}

// Pembuatan dan Inialisasi obyek transformator
$xh = xslt_create();
$fileBase = 'file://' . getcwd () . '/';
xslt_set_base ( $xh, $fileBase );

// Proses transformasi
$result = xslt_process($xh, $xml_file, $xml_file, NULL, array(), $params);
if (!$result) {
    // Terjadi kesalahan pada proses transformasi, tampilkan pesan kesalahan
    echo 'XSLT processing error: ' .xslt_error($xh) ;
}
else {
    // Proses transformasi berhasil, segera cetak file html hasil transformasi
    echo $result;
}

// destroy obyek transformator
xslt_free($xh);

```

Hasil dari proses transformasi program diatas adalah halaman web dengan tampilan sebagai berikut:

Hello Customers

Selamat datang, anda telah berada pada situs pelayanan pelanggan Telkomnet@Home. Di situs ini anda dapat memperoleh informasi tagihan anda pada tiap bulannya, serta informasi detail penggunaan. Di situs ini anda juga dapat melakukan perubahan password akses internet Telkomnet@Home anda secara mudah dan online. Bagi anda yang tidak ingin biaya pemakaian internet anda membengkak, juga tersedia fasilitas seting waktu kontrol akses yang akan membantu anda membatasi penggunaan internet pada tiap satuan waktu yang anda tentukan sendiri.

Selamat menikmati.

What is "telkomnet@home" ?

Rabu, 16 Juni 2004 11:18:57

Personal Detail

User Telkomnet@home : 0315929384@home

Nama : reza

Alamat : Suterejo Utara Prima

Kota : surabaya

Lokasi : Kandatel : SBT
Kancatel : SBT
Unit Pelayanan : MANYAR

Paket : 15 Jam

Phone : 0315929384

Detail Penggunaan

Bulan Tahun

Check your Account everywhere !

Every telkomnet@home account that can be checked everywhere you go!

Account Login !

Gambar 4.2 Tampilan halaman profile pelanggan.

Proses ekstraksi data profile pelanggan dari file XML “cust.xml” dilakukan oleh file “pars_xsl_plg.xml”.

Program 4.3 Program ekstraksi data profile pelanggan (cust.xml) oleh file pars_xsl_plg.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format">
<xsl:output method="html" indent="yes" encoding="utf-8"/>

<xsl:param name="background_color">blue</xsl:param>
<xsl:param name="text_color">#000000</xsl:param>
<xsl:param name="user_home">Empty</xsl:param>

<xsl:template match="/Costumers/costumer[user_home=$user_home]">
  <xsl:call-template name="tablerows"/>
</xsl:template>
```



```

<xsl:template name="tablerows">
  <xsl:for-each select="tablerow">
    .....
    .....<xsl:value-of select="user_home"/>
    .....<xsl:value-of select="nama"/>
    .....<xsl:value-of select="alamat"/>
    .....<xsl:value-of select="kota"/>
    .....<xsl:value-of select="kandatel"/>
    .....<xsl:value-of select="kancatel"/>
    .....<xsl:value-of select="up"/>
    .....<xsl:value-of select="nama_kat"/>
    .....<xsl:value-of select="telepon"/>
    .....
  </xsl:for-each>
</xsl:template>
</xsl:stylesheet>

```

c. Sistem informasi detail penggunaan

Pada halaman profile pelanggan pada bagian bawah, pelanggan dapat memperoleh informasi detail penggunaan pada bulan yang diinginkan beserta rincian biaya yang harus dibayar. Sebagaimana halaman profile pelanggan, halaman detail pelanggan dibuat dengan menggunakan php (http://localhost/home/cust/detail_usage.php), file XML data penggunaan ("C:\Program Files\Apache Group\Apache\htdocs\home\cust\detail\detail[tahun][bulan][username][tahun][bulan].xml"), serta file XSLT (C:\Program Files\Apache Group\Apache\htdocs\home\cust\pars_xsl_detail.xml). Untuk proses transformasi file XML dengan XSLT sama dengan proses transformasi untuk halaman profile, yang berbeda hanya nilai dari variabel \$xml_data dan \$xsl_file. Adapun proses ekstraksi data detail penggunaan pelanggan dapat dijelaskan pada potongan program berikut:

Program 4.4 Program ekstraksi data detail penggunaan pelanggan oleh file pars_xsl_detail.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format">
<xsl:output method="html" indent="yes" encoding="utf-8"/>

<xsl:param name="nama">Empty</xsl:param>
<xsl:param name="nama_kat">Empty</xsl:param>
<xsl:param name="user_home">Empty</xsl:param>

```

```

<xsl:param name="password">Empty</xsl:param>

<xsl:template match="/">
  <xsl:call-template name="tablerows"/>
</xsl:template>

<!-- proses untuk menampilkan informasi Personal detail pelanggan -->
  <xsl:template name="tablerows">
    <xsl:for-each select="Usages">
      .....
      .....<xsl:value-of select="$nama"/>
      .....<xsl:value-of select="$user_home"/>
      .....<xsl:value-of select="translate($nama_kat,'+', ' ')/> Jam
      .....
    </xsl:for-each>
  </xsl:template>

<!-- fungsi untuk memanggil template yang mencetak detail penggunaan -->
  .....<xsl:call-template name="detail_usages"/>

  .....<xsl:value-of select="$biaya_paket"/>
  .....<xsl:value-of select="$biaya_tambahan"/>
  .....<xsl:value-of select="$waktu_tambah"/>
  .....<xsl:value-of select="$biaya_tambahan"/>
  .....<xsl:value-of select="$biaya_total"/>

<!-- proses untuk menampilkan detail penggunaan -->
  <xsl:template name="detail_usages">
    <xsl:for-each select="detail_usage">
      .....
      .....<xsl:value-of select="$num"/>
      .....<xsl:value-of select="$start_time"/>
      .....<xsl:value-of select="$duration_time"/>
      .....
    </xsl:for-each>
  </xsl:template>
</xsl:stylesheet>

```



telkomnet@home - Microsoft Internet Explorer

Address: 0313897114@home&nama=ALI+NAGA+SAPUTRA.&nama_kat=15+Jam&Submit=Submit

Rabu, 16 Juni 2004 9:03:09

Personal Detail

Nama : ALI+NAGA+SAPUTRA.
 User Telkomnet@home : 0313897114@home
 Kategori : 15 Jam

No	Waktu Mulai	Waktu Selesai	Durasi
1	5/11/2004 9:13:05 AM	5/11/2004 9:40:02 AM	1740
2	5/11/2004 10:43:24 AM	5/11/2004 10:46:26 AM	240
3	5/11/2004 2:39:02 PM	5/11/2004 3:15:56 PM	2220
4	5/12/2004 10:01:59 AM	5/12/2004 12:04:39 PM	7500
5	5/12/2004 12:16:02 PM	5/12/2004 12:30:24 PM	900
6	5/12/2004 1:08:46 PM	5/12/2004 1:29:24 PM	1260

Total Waktu Penggunaan :

Perincian Biaya

Biaya Abonemen : Rp.125000,-
 Biaya Tambahan : Rp. 28875,-
 Waktu Tambah : 13860 menit
 Kelebihan Permenit : Rp.28875,-
 Total Biaya : Rp. 26250,-

Gambar 4.3 Tampilan Halaman detail penggunaan.

d. **Sistem seting kontrol waktu akses**

Selain dapat memperoleh informasi detail penggunaan dan biaya tagihan, pelanggan juga dapat melakukan seting kontrol waktu akses. Pelanggan dapat menentukan sendiri lama waktu maksimal perharinya untuk melakukan akses Internet. Pada saat pelanggan melakukan seting kontrol waktu akses, sistem akan menambahkan data pada tabel **RadReply**.

Program 4.5 Program untuk menuliskan nilai kontrol waktu akses dari pelanggan ke dalam tabel RadReply

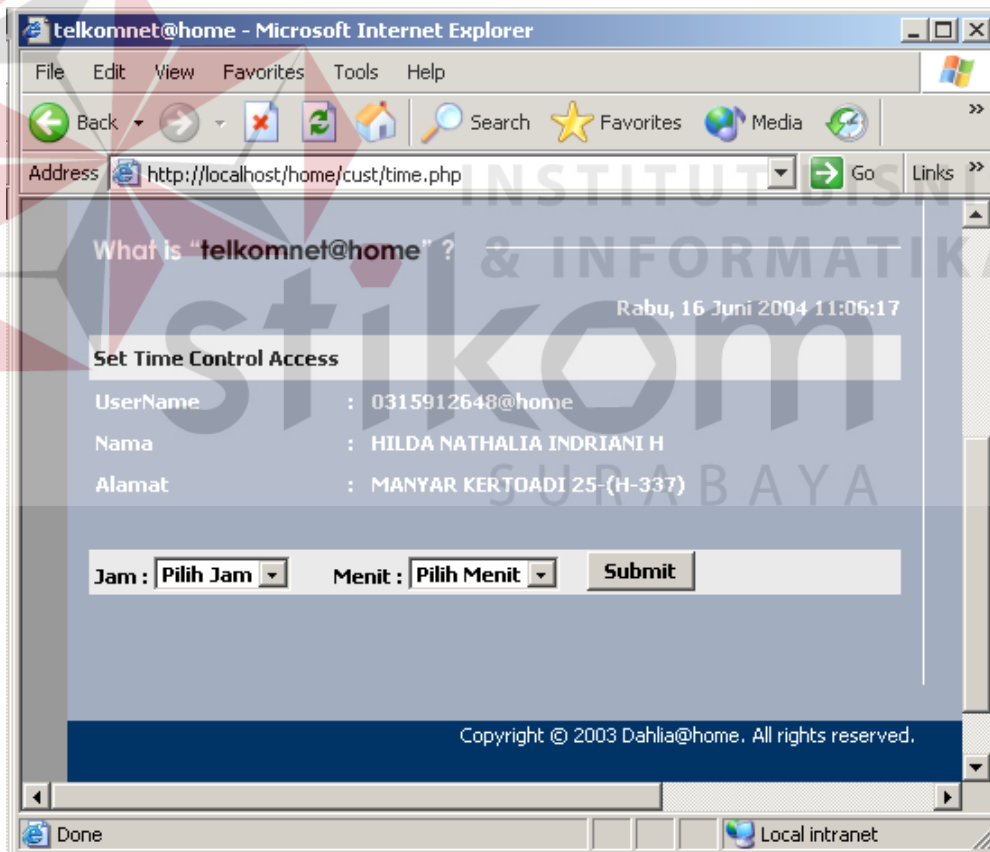
```
if ($Submit=="Submit") {
    //Hitung nilai variabel kontrol waktu akses dalam satuan detik
    $detik= $jam*3600+$menit*60;
    //cek apakah sebelumnya user sudah pernah melakukan setting kontrol waktu akses
```

```

$query = "select count(id) as id_radreply from radreply where username='$user_home'
and attribute='time'";
dbquery($con,$query,$stmt0,$d);
$хid_radreply= (dbresult(dbfetch($stmt0),"id_radreply"));

if ($хid_radreply==0){
    // Jika belum, maka lakukan operasi INSERT data pada tabel RadReply
    $query = "select max(id) as maxid from radreply";
    dbquery($con,$query,$stmt1,$d);
    $maxId= dbresult($stmt1,"maxid");
    if (!(isset($maxId)))
        $maxId= 0;
    $query = "insert into radreply values($maxId,'$user_home','time','','$detik')";
    dbquery($con,$query,$stmt1,$d);
}else{
    // Jika belum, maka lakukan operasi UPDATE data pada tabel RadReply
    $query = "update radreply set value='$detik' where userNаme='$user_home'
and attribute='time'";
    dbquery($con,$query,$stmt1,$d);
}
}
}

```



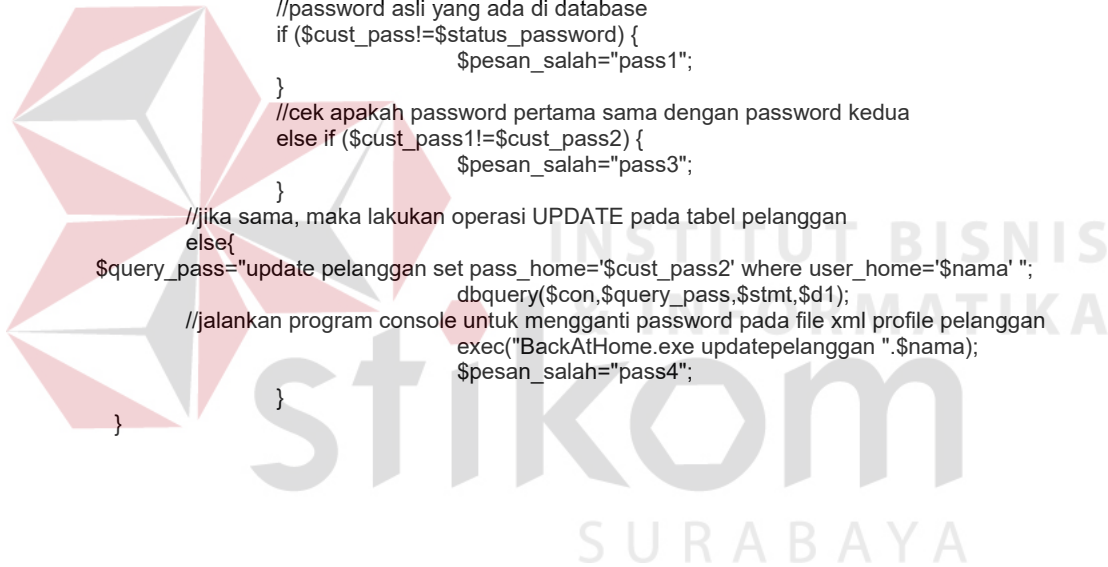
Gambar 4.4 Tampilan halaman seting kontrol waktu akses.

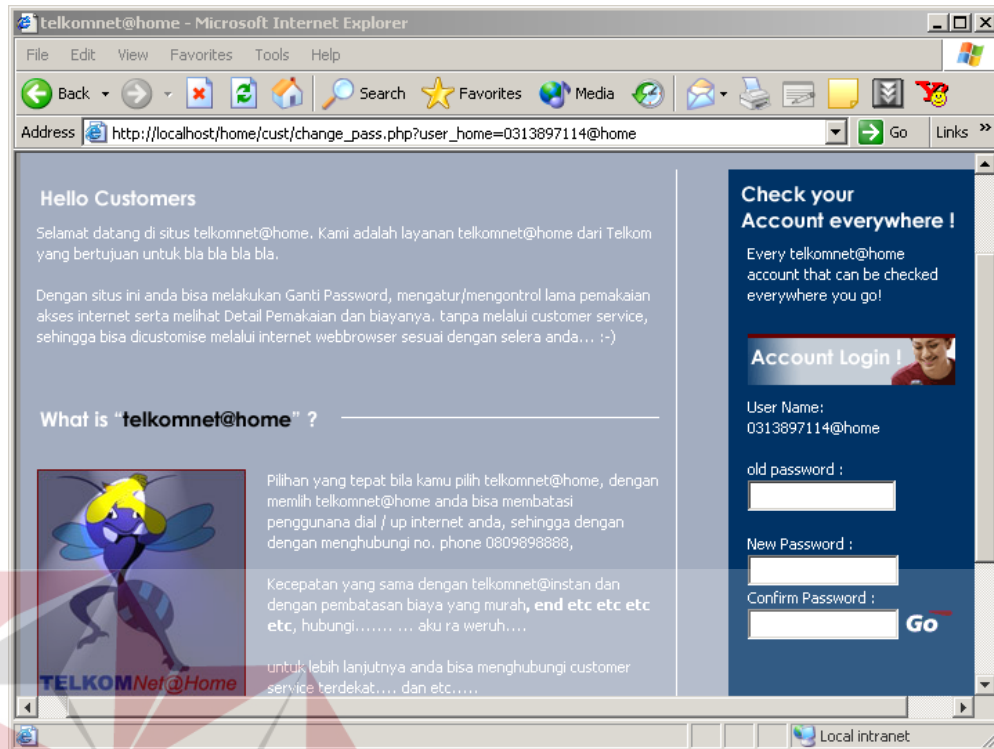
e. Sistem ganti password

Pelanggan dapat mengganti password akses Internet TELKOMNet@Home sekaligus password untuk mengakses situs web pelayanan eksternal dengan memilih menu ganti password. Sistem secara langsung akan mengubah password pelanggan baik pada tabel pelanggan maupun file XML data profile pelanggan. Berikut adalah potongan program untuk mengganti password.

Program 4.6 Program untuk ganti password

```
// Cek apakah sedang melakukan aktifitas penggantian password
if (isset($status_loginku)) {
    //cek apakah password asli yg diinputkan pelanggan sama dengan
    //password asli yang ada di database
    if ($cust_pass!=$status_password) {
        $pesan_salah="pass1";
    }
    //cek apakah password pertama sama dengan password kedua
    else if ($cust_pass1!=$cust_pass2) {
        $pesan_salah="pass3";
    }
    //jika sama, maka lakukan operasi UPDATE pada tabel pelanggan
    else{
        $query_pass="update pelanggan set pass_home='$cust_pass2' where user_home='$nama' ";
        dbquery($con,$query_pass,$stmt,$d1);
        //jalankan program console untuk mengganti password pada file xml profile pelanggan
        exec("BackAtHome.exe updatepelanggan ".$nama);
        $pesan_salah="pass4";
    }
}
```





Gambar 4.5 Tampilan halaman ganti password.

4.2.2 Sistem aplikasi internal

Sistem aplikasi internal memiliki tiga tingkatan pengguna, yaitu Administrator, Customer Care dan Managemen. Masing-masing memiliki hak yang berbeda-beda dalam menjalankan sistem. Untuk menggunakan aplikasi internal, pengguna aplikasi dapat mengunjungi alamat *http://radius.home.net*

a. Implementasi daftar user Administrator

Sistem Administrator memiliki hak untuk melakukan manajemen terhadap para user Administrator yang meliputi Administrator sendiri, Customer Care dan Manajemen. Manajemen user yang dimaksud meliputi menambah, menghapus, mengedit daftar user manajemen.

Kamis, 17 Jun 2004 10:44:02		Welcome a you as by System Administrator				
Untuk menambah User Administrator Klik disini!						
USER NAME	PASSWORD	LEVEL	TELEPON	EMAIL	EDIT	DELETE
kia	admin	System Administrator	08565458	kia@telkom.net	Edit	delete
edi	mgm	Management	081234567	edi@telkom.co.id	Edit	delete
tia	ccare	Customer Care	08159753	tia@telkom.net	Edit	delete

Gambar 4.6 Tampilan Halaman daftar user Administrator.

Form Input Web Administrator TELKOMNet@Home

User Name	<input type="text" value="kia"/>
Password	<input type="text" value="admin"/>
Level	<input type="text" value="System Administrator"/>
Nama	<input type="text" value="Kia"/>
Telepon	<input type="text" value="08565458"/>
Email	<input type="text" value="kia@telkom.net"/>
Loker	<input type="text" value="sby"/>
Keterangan	<input type="text"/>

Gambar 4.7 Tampilan form input web Administrator.

b. Implementasi untuk mendapatkan daftar pelanggan aktif

Customer Care memiliki tugas untuk menganalisa kondisi para pelanggan untuk kemudian memberikan pelayanan sesuai dengan prosedur yang telah ditetapkan, salah satunya adalah pelanggan aktif. Untuk dapat melihat pelanggan mana saja yang masuk kategori pelanggan aktif, Customer Care dapat membuka halaman daftar pelanggan aktif. Sistem akan mengambil data dari file **cust.xml**

(C:\Program Files\Apache Group\Apache\htdocs\home\cust\cust.xml) yang kemudian akan digabungkan dengan file **pars_pelanggan_aktif.xsl** (C:\Program Files\Apache Group\Apache\htdocs\home\admin\pelanggan\pars_pelanggan_aktif.xsl) oleh **new_pelanggan_aktif.php** (C:\Program Files\Apache Group\Apache\htdocs\home\admin\pelanggan\pars_pelanggan_aktif.xsl). Untuk proses transformasi oleh file php dapat digambarkan oleh petikan program berikut:

Program 4.7 Program untuk transformasi data pelanggan aktif

```

$xml_file = "../cust/cust.xml";
$xml_file = "pars_pelanggan_aktif.xsl";

//agar file xslt dapat mengambil nilai variabel yang dikirim oleh file php yang memanggilnya,
// perlu dilakukan proses pooling nilai variabel tersebut dalam variabel array untuk kemudian
//dikirim ke file xslt melalui parameter fungsi xslt_process
$vars = explode("&", $_SERVER['QUERY_STRING']);
for ($i=0;$i<=count($vars);$i++) {
    $svar = explode("=", $vars[$i]);
    $params[$svar[0]] = $svar[1];
}

// Pembuatan dan Inisialisasi obyek transformator
$xh = xslt_create();
$fileBase = 'file:/' . getcwd() . '/';
xslt_set_base ($xh, $fileBase);

// Proses transformasi
$result = xslt_process($xh, $xml_file, $xsl_file, NULL, array(), $params);
if (!$result) {
    // Terjadi kesalahan pada proses transformasi, tampilkan pesan kesalahan
    echo "XSLT processing error: " . xslt_error($xh);
} else {
    // Proses transformasi berhasil, segera cetak file html hasil transformasi
    echo $result;
}

// destroy obyek transformator
xslt_free($xh);

```

Sedangkan proses ekstraksi informasi data para pelanggan yang masuk dalam kategori pelanggan aktif dilakukan oleh file XSLT dengan potongan program sebagai berikut:

Program 4.8 Program untuk mengambil data pelanggan aktif dari file cust.xml

```

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format">
<xsl:output method="html" indent="yes" encoding="utf-8"/>

```



```

//inisialisasi variabel GET HTTP sesuai dengan yang dikirimkan oleh file php yang memanggil
<xsl:param name="index_page">1</xsl:param>
<xsl:param name="num_rec">10</xsl:param>
<xsl:param name="nilai"></xsl:param>

<xsl:variable name="upper" select="$index_page * $num_rec" />
<xsl:variable name="lower" select="(($index_page -1)* $num_rec)+1" />
<xsl:variable name="backgr">#DBDFE0</xsl:variable>

//fungsi untuk memanggil template dengan nama mine
<xsl:template match="/">
    <xsl:call-template name="mine"/>
</xsl:template>

<xsl:template name="mine">
    .....
    .....
    //fungsi untuk menandatangani node costumer dengan syarat ber-status aktifasi=1
    //dan nilai user_home yang diinputkan oleh pelanggan dalam kotak input search.
    <xsl:for-each select="Costumers/costumer[(status_aktifasi=1 and contains(user_home,$nilai))]">
    <xsl:choose>

//fungsi untuk menampilkan node dengan posisi antara batas bawah dan batas atas
    <xsl:when test="((position() &gt; ($lower))or(position()= ($lower))) and ((position() &lt; $upper) or
    (position()= $upper))">
    <xsl:choose>

//fungsi untuk dapat mencetak node costumer dalam baris yang beda warna dan berselang seling.
    <xsl:when test="position() mod 2= 0" >
    <tr bgcolor="#DBDFE3">
        .....
    </tr>
    </xsl:when>
    <xsl:otherwise>
    <tr bgcolor="#F0F0F0">
        .....
    </tr>
    </xsl:otherwise>
    </xsl:choose>
    </xsl:when>
    </xsl:choose>
    </xsl:for-each>
    .....
</xsl:template>
</xsl:stylesheet>

```

Adapun halaman pelanggan aktif yang didapat dari program di atas adalah sebagai berikut:

DAFTAR PELANGGAN AKTIF

SEARCH @home Search Untuk menambah Pelanggan Baru [Klik Disini!](#)

Num	NAMA	ALAMAT	KOTA	TELEPON	EMAIL	USER HOME	PASS HOME	PAKET	DEAKTIFASI
1	reza	Suterejo Utara Prima	surabaya	0315929384	reza@yahoo.com	0315929384@home	hore	15 Jam	Deaktifasi
2	RUDYANTO HANDAJAJA	SATELIT UTARA 9 BL JT 11	SURABAYA	0317313875	-	0317313875@home	28121977	15 Jam	Deaktifasi
3	rossi	kendangsari	surabaya	0318414232	rossilazuardi@telkom.net	0318414232@home	gembira	15 Jam	Deaktifasi
4	ALI NAGA SAPUTRA	KARANG EMPAT BESAR 76	SURABAYA	0313897114	-	0313897114@home	171001	15 Jam	Deaktifasi
5	WIDJAYA HADI PRANOTO	DIAN ISTANA BL D2 NO. 12	SURABAYA	0317527557	lita_n@hotmail.com	0317534343@home	litus	50 Jam	Deaktifasi
6	TEGUH PRASETYO	MANYAR JAYA 8 / C-8	SURABAYA	0315941477/70993968	-	0315925605@home	fiona	15 Jam	Deaktifasi

Gambar 4.8 Tampilan Halaman daftar pelanggan aktif.

c. Implementasi untuk menambah pelanggan

Customer Care memiliki hak untuk melakukan penambahan data pelanggan baru. Hal ini dilakukan dengan meng-klik menu tambah user lalu mengisikan data pelanggan baru tersebut pada browser pop up yang muncul seperti berikut:

The image shows a screenshot of a web browser window titled "INPUT PELANGGAN BARU : - Microsoft Internet Explorer". The main content is a form titled "Form Input Pelanggan Baru TELKOMNet@Home". The form has a light purple background and contains the following fields:

- Nama: Text input field
- Alamat: Text input field
- Kota: Text input field
- Telepon: Text input field
- Email: Text input field
- User TelkomNet@Home: Text input field with "@home" suffix
- Lokasi: A group of three text input fields labeled "Kandatel", "Kancatel", and "Unit Pelayanan"
- Password TelkomNet@Home: Text input field
- Kategori: Dropdown menu showing "15 Jam"
- Tanggal Registrasi: Text input field showing "18-06-2004 09:51:48"
- Tanggal Aktifasi: Text input field showing "18-06-2004 09:51:48" with a calendar icon

At the bottom of the form are two buttons: "SUBMIT" and "RESET".

Gambar 4.9 Tampilan halaman pop up untuk penambahan user.

Proses penambahan dilakukan setelah Customer Care menekan tombol submit. Proses penambahan data pelanggan baru dilakukan dengan menambahkan data baru tersebut pada **tabel pelanggan** dan file XML profile pelanggan (**cust.xml**). Untuk pelanggan yang memilih jadwal aktifasi pada hari setelah hari dia mendaftar, maka data pelanggan akan dimasukkan dengan nilai `status_aktifasi= 0`, yang berarti pelanggan sedang dalam proses aktifasi, namun jika waktu aktifasi menunjuk pada hari itu juga, maka secara langsung pelanggan akan dikategorikan sebagai pelanggan aktif dengan nilai `status_aktifasi= 1`. Berikut adalah program yang dijalankan untuk melakukan proses tersebut:

Program 4.9 Program untuk menambah data pelanggan baru

```
.....
include_once "../include/connect.php";
.....
```

```

//proses pengecekan apakah pelanggan baru menggunakan userhome yang telah ada
$query3="select * from pelanggan where user_home='$userhome'";
dbquery($con,$query3,$stmt3,$d3);
$stmta3=dbfetch($stmt3);

//Jika pelanggan baru menggunakan userhome yang belum terpakai, maka data dapat
//segera diproses
if (mysql_num_rows(mysql_query($query3))==0) {
    $vtanggal=substr($schedule_datetime,0,2);$vbulan=substr($schedule_datetime,3,2);
    $vtahun=substr($schedule_datetime,6,4);
    $mday=substr($tanggal1,0,2);$month=substr($tanggal1,3,2);$year=substr($tanggal1,6,4);
    $vschedule=$vtahun."-".$vbulan."-".$vtanggal;
    $timestamp1 = mktime(0,0,0,$month,$mday,$year);
    $timestamp2 = mktime(0,0,0,$vbulan,$vtanggal,$vtahun);
    $status_aktifasi=0;
    //timestamp2 adalah variabel datetime yang berisi jadwal aktifasi yang diinginkan
    //timestamp1 adalah variabel datetime yang berisi waktu saat ini
    //Jika waktu jadwal aktifasi yang diinginkan terjadi pada hari esok maka data pelanggan
    //dimasukkan dengan status_aktifasi=0, sebaliknya nilai status_aktifasi bernilai 1.
    if ($timestamp2 <= $timestamp1) {
        .....//buat perintah SQL insert dengan status aktifasi=1 pada variabel $query
    }else{
        .....// buat perintah SQL insert dengan status aktifasi=0 pada variabel $query
    }
    .....
    //Proses penambahan node data pelanggan baru pada file cust.xml
    exec("BackAtHome.exe addpelanggan ".$userhome);
    .....
}

```

d. Implementasi untuk mendapatkan daftar pelanggan dalam proses

Proses pengambilan daftar pelanggan dalam proses tidak berbeda jauh dengan halaman pelanggan aktif pada penjelasan sebelumnya. Untuk mendapatkan daftar pelanggan dalam proses, baik proses aktifasi ataupun proses deaktivasi, sistem cukup mengganti syarat pengambilan data pelanggan pada halaman daftar pelanggan aktif dari status_aktifasi=1 menjadi status_aktifasi=0.

Berikut adalah potongan program yang menjelaskan hal tersebut:

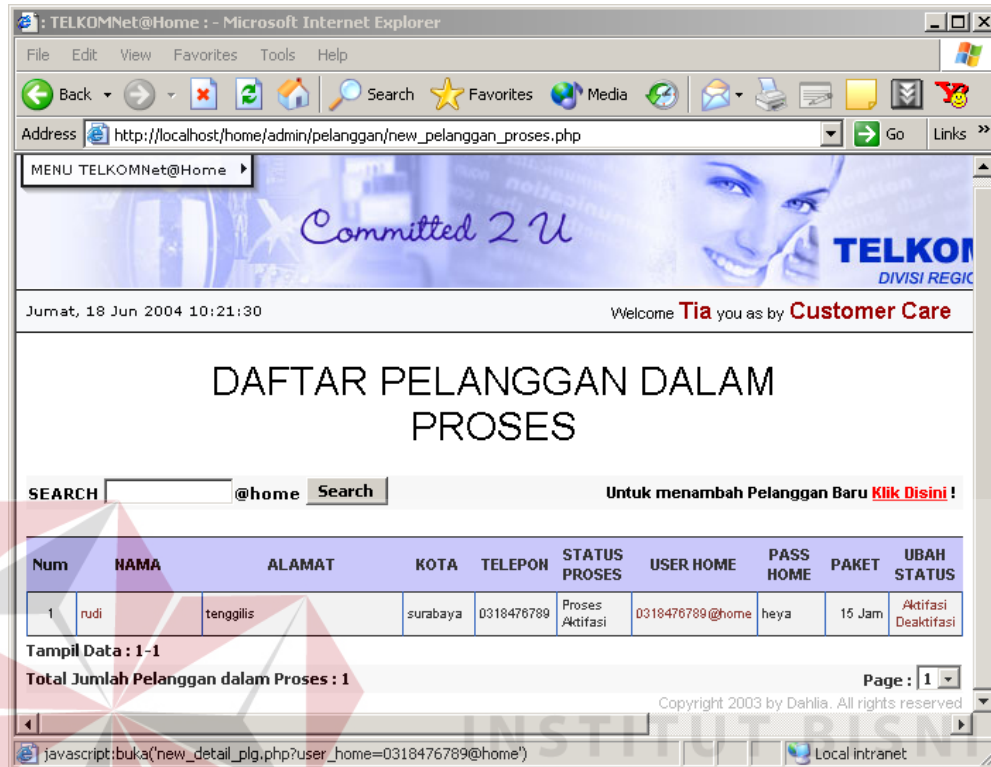
Program 4.10 Program untuk mengambil data pelanggan dalam proses

```

.....
<xsl:for-each select="Costumers/costumer[status_aktifasi=0 and contains(user_home,$nilai)]">
<xsl:variable name="user_home"><xsl:value-of select="substring-before(user_home,'@home')"/>
</xsl:variable>
<xsl:choose>
<xsl:when test="((position() > ($lower))or(position()= ($lower))) and ((position() &lt; $upper) or
(position()= $upper))">
<xsl:choose>
<xsl:when test="position() mod 2= 0" >
<tr bgcolor="#DBDFE3">
.....
.....

```

Halaman pelanggan dalam proses yang dihasilkan adalah seperti berikut:



Gambar 4.10 Tampilan halaman daftar pelanggan dalam proses.

e. Implementasi untuk mendapatkan daftar pelanggan non aktif

Seperti halnya halaman pelanggan aktif dan dalam proses. Halaman daftar pelanggan non aktif juga didapat dengan cara yang sama, yang berbeda hanya status_aktifasi dari pelanggan, untuk mendapatkan pelanggan non aktif, sistem akan mengambil data pelanggan dalam file cust.xml yang memiliki nilai status_aktifasi= 2. Berikut adalah potongan program yang menunjukkan hal tersebut.

Program 4.11 Program untuk mengambil data pelanggan non aktif

```
.....
<xsl:for-each select="Costumers/costumer[status_aktifasi=2 and contains(user_home,$nilai)]">
<xsl:variable name="user_home"><xsl:value-of select="substring-before(user_home,'@home')"/></xsl:variable>
<xsl:choose>

```

```

<xsl:when test="((position() > ($lower))or(position)= ($lower)) and ((position() < $upper) or
(position)= $upper)">
<xsl:choose>
  <xsl:when test="position() mod 2= 0" >
    <tr bgcolor="#DBDFE3">
      .....
    .....
    .....
  
```

Halaman daftar pelanggan non aktif ditunjukkan seperti gambar berikut:

The screenshot shows a web browser window with the following content:

- Address bar: `http://localhost/home/admin/pelanggan/new_pelanggan_non_aktif.php`
- Page Header: "Committed 2 U" and "TELKOM DIVISI REGIO"
- Date/Time: "Jumat, 18 Jun 2004 10:26:34"
- Welcome message: "Welcome Tia you as by Customer Care"
- Section Title: "DAFTAR PELANGGAN NON AKTIF"
- Search bar: "SEARCH [input] @home Search"
- Text: "Untuk menambah Pelanggan Baru [Klik!](#)"
- Table of inactive customers:

Num	NAMA	ALAMAT	KOTA	TELEPON	EMAIL	USER HOME	PASS HOME	PAKET	AKTIFASI
1	Jay	JL Rungkut Mejoyo Selatan	Surabaya	0318437281	jay@yahoo.com	0318437281@home	hello	15 Jam	Aktifasi
2	Ihsan	ketintang	surabaya	0318290331	ihsan@pln.co.id	0318290331@home	hore	15 Jam	Aktifasi

Summary: "Tampil Data : 1-2", "Total Jumlah Pelanggan Non Aktif : 2", "Page : 1"

Gambar 4.11 Tampilan halaman daftar pelanggan non aktif.

f. Implementasi untuk mendapatkan daftar pelanggan penggunaan nol

Daftar pelanggan penggunaan nol mengacu pada para pelanggan yang dalam bulan tertentu belum pernah sama sekali melakukan akses Internet TELKOMNet@Home. Hal ini penting sekali bagi Customer Care untuk menindaklanjuti guna memperoleh informasi apakah telah terjadi sesuatu pada sistem yang menyebabkan pelanggan tidak memakai akses Internet, hal ini

berkaitan dengan upaya untuk mempertahankan kualitas pelayanan akses Internet. Data pelanggan yang belum pernah menggunakan akses Internet dapat dilihat pada file **cust.xml**, dimana setiap penggunaan Internet setiap bulannya tercatat sebagai node “**call[tahun][bulan]**” yang menunjukkan berapa kali pelanggan melakukan dial ke TELKOMNet@Home, dan node “**cost[tahun][bulan]**” yang menunjukkan biaya yang harus ditanggung pelanggan pada tahun dan bulan yang dimaksud. Sehingga untuk mendapat pelanggan yang belum pernah melakukan akses Internet, cukup dengan mencari node customer pada file cust.xml yang tidak memiliki node anak “**call[tahun][bulan]**” dan “**cost[tahun][bulan]**”. Berikut adalah potongan program yang menunjukkan proses tersebut:

Program 4.12 Program untuk mengambil data pelanggan penggunaan nol

```

.....
<xsl:for-each select="Customers/customer[(status_aktifasi=1 and
contains(user_home,$nilai)) and(count(*[name()='concat('call',$tahun,$bulan)])=0)]">
<xsl:choose>
  <xsl:when test="((position() > ($lower)) or (position() = ($lower))) and ((position() &lt; $upper) or
(position() = $upper))">
    <xsl:variable name="durasidtk"><xsl:value-of select="*[name()='concat('call',$tahun,$bulan)]"
/></xsl:variable>
    <xsl:choose>
      <xsl:when test="position() mod 2= 0" >
        <tr bgcolor="#DBDFE3">
          .....
          .....
          .....

```

g. Implementasi untuk mendapatkan detail info tagihan

Untuk mendapatkan detail info tagihan pelanggan tertentu, Customer Care dapat memilih menu info tagihan, kemudian memilih detail untuk melihat detail info tagihan dari pelanggan yang diinginkan. Adapun informasi detail tagihan diperoleh dari file XML detail penggunaan ([username]_[tahun][bulan].xml). Berikut adalah potongan program yang menjelaskan bagaimana ekstraksi data pada file XML detail penggunaan dilakukan:

Program 4.13 Program untuk mengambil data detail info tagihan

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format">
<xsl:output method="html" indent="yes" encoding="utf-8"/>

//karena informasi nama, user_home dan jenis paket tidak terdapat pada file xml
//detail pelanggan, maka informasi tersebut didapat dari variabel GET HTTP request
//yang dikirim oleh halaman php yang memanggilnya, yakni new_detail_billing.php
<xsl:param name="nama">Empty</xsl:param>
<xsl:param name="user_home">Empty</xsl:param>
<xsl:param name="paket">Empty</xsl:param>

<xsl:template match="/">
    <xsl:call-template name="mine"/>
</xsl:template>
.....
//proses mengambil informasi dari variabel GET HTTP
.....<xsl:value-of select="translate($nama,'%20',' ')/>
.....<xsl:value-of select="$user_home"/>
.....<xsl:value-of select="substring-before($paket,'%20Jam')"/>

//proses mengambil informasi dari node pada file xml detail penggunaan
.....Rp. <xsl:value-of select="Usages/biaya_paket"/>,-
.....Rp. <xsl:value-of select="Usages/biaya_lebih_permenit"/>,-
.....<xsl:value-of select="floor(Usages/waktu_tambah div 60)"/>
.....Rp. <xsl:value-of select="Usages/biaya_tambahan"/>,-
.....Rp. <xsl:value-of select="Usages/biaya_total"/>,-
.....
</xsl:template>
</xsl:stylesheet>

```

Halaman detail info tagihan pelanggan hasil dari program diatas dapat ditunjukkan seperti berikut:



Detail Billing Pelanggan TELKOM Net@Home	
Nama	: HARIJANTO SOEPRAPTO
User TelkomNet@Home	: 0315930671@home
Kategori	: 15 Jam
Biaya Abonemen	: Rp. 125000,-
Kelebihan per menit	: Rp. 125,- per menit
Waktu Tambah	: 185 menit
Biaya Tambah	: Rp. 23125,-
Total Biaya	: Rp. 22750,-

Gambar 4.12 Tampilan halaman detail info tagihan

h. Implementasi untuk mendapatkan info detail penggunaan

Untuk dapat memperoleh informasi detail penggunaan seorang pelanggan, Customer Care terlebih dahulu membuka halaman daftar pelanggan aktif beserta resume total waktu penggunaan pada bulan tertentu. Baru kemudian Customer Care dapat memilih menu detail untuk melihat detail penggunaan seorang pelanggan.

Program 4.14 Program untuk mengambil data detail penggunaan

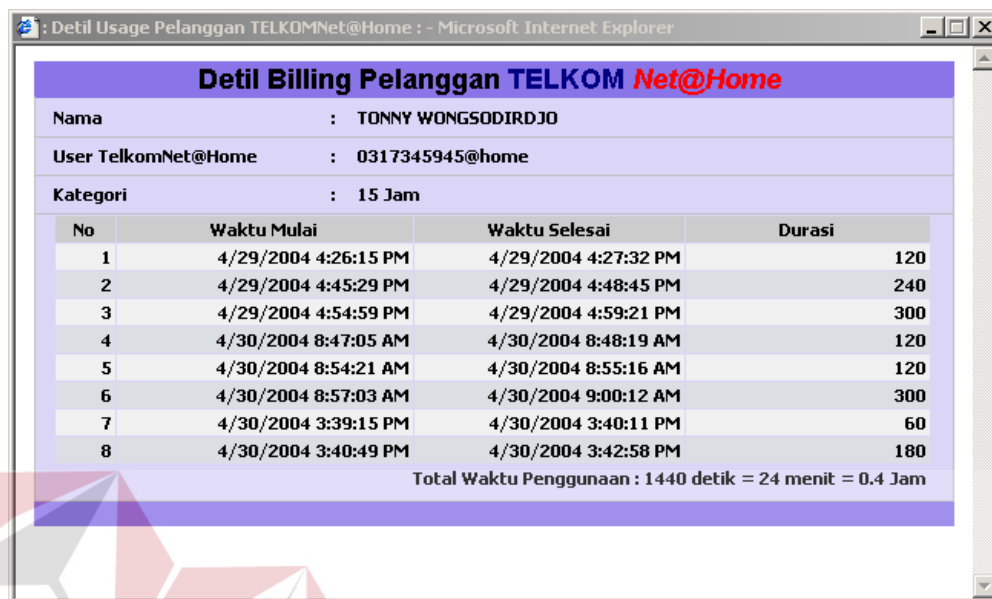
```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format">
<xsl:output method="html" indent="yes" encoding="utf-8"/>

//karena informasi nama, user_home, paket tidak terdapat pada file xml
//detail pelanggan, maka informasi tersebut harus didapat dari file php
//yang memanggilnya sebagai variabel GET HTTP request
<xsl:param name="nama">Empty</xsl:param>
<xsl:param name="user_home">Empty</xsl:param>
<xsl:param name="paket">Empty</xsl:param>
<xsl:param name="exist">no</xsl:param>

<xsl:template match="">
  <xsl:call-template name="mine"/>
</xsl:template>

<xsl:template name="mine">
  .....
  //proses mengambil informasi dari variabel GET HTTP request
  .....<xsl:value-of select="translate($nama,'%20',' ')" />
  .....<xsl:value-of select="$user_home" />
  .....<xsl:value-of select="substring-before($paket,'%20Jam')" />
  .....
  //proses mengambil informasi dari file xml detail penggunaan
  <xsl:for-each select="Usages/detail_usage">
    .....<xsl:value-of select="position()" />
    .....<xsl:value-of select="start_time" />
    .....<xsl:value-of select="end_time" />
    .....<xsl:value-of select="duration_time" />
  </xsl:for-each>
  .....
</xsl:template>
</xsl:stylesheet>
```

Halaman detil penggunaan dari salah satu pelanggan ditunjukkan seperti berikut:



Detil Billing Pelanggan TELKOM Net@Home			
Nama	: TONNY WONGSODIRDJO		
User TelkomNet@Home	: 0317345945@home		
Kategori	: 15 Jam		
No	Waktu Mulai	Waktu Selesai	Durasi
1	4/29/2004 4:26:15 PM	4/29/2004 4:27:32 PM	120
2	4/29/2004 4:45:29 PM	4/29/2004 4:48:45 PM	240
3	4/29/2004 4:54:59 PM	4/29/2004 4:59:21 PM	300
4	4/30/2004 8:47:05 AM	4/30/2004 8:48:19 AM	120
5	4/30/2004 8:54:21 AM	4/30/2004 8:55:16 AM	120
6	4/30/2004 8:57:03 AM	4/30/2004 9:00:12 AM	300
7	4/30/2004 3:39:15 PM	4/30/2004 3:40:11 PM	60
8	4/30/2004 3:40:49 PM	4/30/2004 3:42:58 PM	180
Total Waktu Penggunaan : 1440 detik = 24 menit = 0.4 Jam			

Gambar 4.13 Tampilan Halaman info detil penggunaan.

i. Sistem laporan pertumbuhan jumlah pengguna

Penggunaan sistem aplikasi internal juga ditujukan untuk kepentingan manajemen guna mendapatkan laporan pertumbuhan jumlah pengguna, pertumbuhan penggunaan dan pertumbuhan jumlah panggilan. Ketiga jenis laporan tersebut mengambil data dari file XML laporan per tahun (report[tahun][bulan].xml). Berikut adalah potongan program yang menunjukkan bagaimana proses mencetak laporan pertumbuhan jumlah pengguna.

Program 4.15 Program untuk mencetak laporan pertumbuhan jumlah pengguna

```
//proses seleksi node yang memiliki anak dengan atribu data = 'pelanggan't
<xsl:for-each select="reports/bulan/*[@data='pelanggan']/parent::*">
<xsl:choose>
  <xsl:when test="position() mod 2= 0" >
    <tr bgcolor="#DBDFE3">
      .....
      //proses mencetak nama bulan
      <xsl:value-of select="@nama"/>
      //proses untuk mencetak jumlah pelanggan baru untuk tiap kandatelnya
      <xsl:value-of select="kandatel[@nama='SBB' and @data='pelanggan']/@jumlah"/>
      <xsl:value-of select="kandatel[@nama='SBT' and @data='pelanggan']/@jumlah"/>
    </tr>
  </xsl:when>
</xsl:choose>
</xsl:for-each>
```

```

<xsl:value-of select="kandatel[@nama='MALANG' and @data='pelanggan']/@jumlah"/>
<xsl:value-of select="kandatel[@nama='MADIUN' and @data='pelanggan']/@jumlah"/>
<xsl:value-of select="kandatel[@nama='JEMBER' and @data='pelanggan']/@jumlah"/>
//proses untuk mencetak total pertumbuhan jumlah pelanggan pada bulan tertentu
<xsl:value-of select="sum(kandatel[@data='pelanggan']/@jumlah)"/>
.....
</xsl:when>
<xsl:otherwise>
  <tr bgcolor="#F0F0F0">
    .....
    <xsl:value-of select="@nama"/>
    <xsl:value-of select="kandatel[@nama='SBB' and @data='pelanggan']/@jumlah"/>
    <xsl:value-of select="kandatel[@nama='SBT' and @data='pelanggan']/@jumlah"/>
    <xsl:value-of select="kandatel[@nama='MALANG' and @data='pelanggan']/@jumlah"/>
    <xsl:value-of select="kandatel[@nama='MADIUN' and @data='pelanggan']/@jumlah"/>
    <xsl:value-of select="kandatel[@nama='JEMBER' and @data='pelanggan']/@jumlah"/>
    <xsl:value-of select="sum(kandatel[@data='pelanggan']/@jumlah)"/>
    .....
  </xsl:otherwise>
</xsl:choose>
</xsl:for-each>

```

Adapun proses pembuatan grafik dilakukan dengan cara menggunakan file **user_graph.php** yang telah dimodifikasi untuk menampilkan grafik dengan menggunakan parameter-parameter yang terdiri atas:

1. ytitle, judul dari garis Y pada grafik
2. count_month, jumlah bulan yang datanya akan ditampilkan(dalam hal ini 4)
3. current_month, bulan aktif saat ini
4. current_year, tahun aktif saat ini
5. data pelanggan baru untuk setiap bulan per kandatelnya, adapun format yang digunakan adalah [nama_kandatel]_[bulan]=jumlah_pelanggan_baru.

Berikut adalah potongan program yang menunjukkan proses penyusunan parameter guna mendapatkan grafik pertumbuhan jumlah pengguna sesuai dengan yang diinginkan.

Program 4.16 Program untuk mendapatkan grafik pertumbuhan jumlah pengguna

```

.....
//proses inialisasi variabel ytitle,count_month, current_month
paramstr= "ytitle=Juml. Pengguna&amp;count_month=4&amp;current_month="+<xsl:value-of
select="$current_month"/>+"&amp; ";
paramstr= paramstr+"current_year="+<xsl:value-of select="$current_year"/>;
.....
//proses seleksi node kandatel yang memiliki nilai atribut data = 'pelanggan' untuk mendapatkan
//nilai pertumbuhan pelanggan baru setiap bulan per tiap bulannya.
<xsl:for-each select="//kandatel[@data='pelanggan']">
  <script>

```

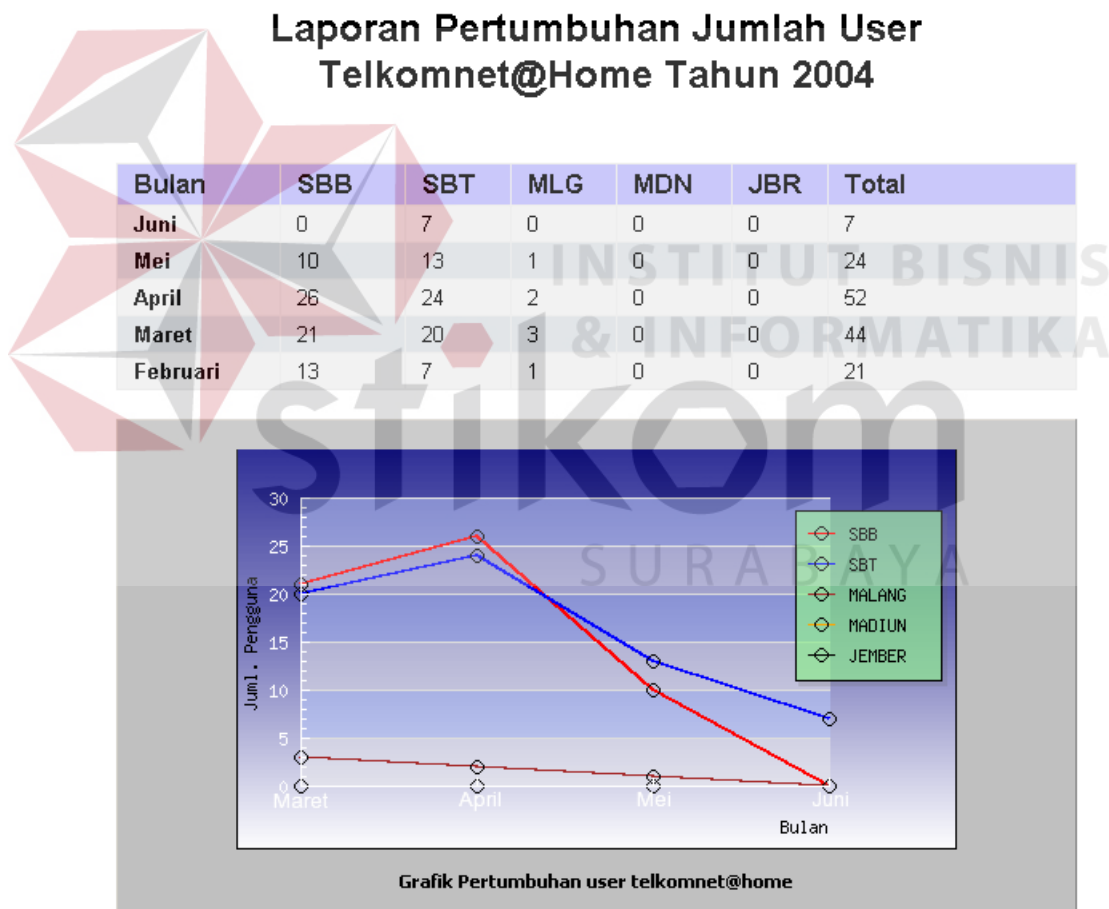
```

paramstr=paramstr+"&"+<xsl:value-of select="@nama"/>+"_"<xsl:value-of
select="@bulan"/>+"="<xsl:value-of select="@jumlah"/>";
</script>
</xsl:for-each>
<script>

//proses penulisan sintak html <img src... dengan menggunakan tag javascript.
document.write("&lt;img src='user_graph.php?'+paramstr+' border='0' align='center' width='450'
height='250' /&gt;");
</script>
.....

```

Adapun hasil dari program diatas dapat dilihat pada halaman laporan pertumbuhan pengguna seperti ditunjukkan pada Gambar 4.14.



Gambar 4.14 Tampilan halaman Laporan jumlah pengguna

g. Sistem laporan pertumbuhan jumlah panggilan

Pembuatan laporan pertumbuhan jumlah panggilan tidak banyak perbedaan dengan proses pembuatan laporan pertumbuhan jumlah pengguna. Yang membedakan hanya pada proses seleksi node pada file laporan per tahun. Jika pada pembuatan laporan pertumbuhan jumlah pengguna node yang dipilih adalah node kandatel dengan atribut data bernilai 'pelanggan', namun pada laporan penggunaan yang dipilih adalah node kandatel yang memiliki nilai atribut data sama dengan 'penggunaan'. Begitu juga pada proses pembuatan grafiknya.

h. Sistem laporan pertumbuhan penggunaan

Sistem laporan pertumbuhan durasi pemakaian juga sama dengan proses pembuatan dua laporan sebelumnya, yang membedakan hanya proses seleksi node kandatelnya saja. Untuk durasi, nilai atribut data adalah 'durasi'.

i. Sistem analisa laporan pertumbuhan

Proses analisa pertumbuhan baik pada angka pertumbuhan jumlah pengguna, pertumbuhan jumlah panggilan dan pertumbuhan penggunaan dilakukan secara bersamaan, dimana setiap angka pertumbuhan diasumsikan terdapat dua kondisi perubahan, yaitu penurunan angka. Dengan demikian akan didapat 3^2 kemungkinan kondisi. Dimana setiap kondisi tersebut memiliki aspek penyebab dan tindakan yang telah ditetapkan sebelumnya sebagai bagian dari kebijakan manajemen. Proses untuk dapat menghasilkan analisa mengenai sebab dan tindakan dilakukan dengan cara mengumpulkan semua aspek sebab dan tindakan semua kemungkinan kondisi yang ada ke dalam variabel array, untuk kemudian dipilih variabel yang mana yang sesuai dengan kondisi yang terjadi

pada bulan tersebut. Berikut adalah potongan program yang menunjukkan bagaimana proses tersebut dilakukan:

Program 4.17 Program untuk mendapatkan Analisa grafik pertumbuhan

```
function analisaSebabTindakan(kondisi)
{
    analisa_sebab= Array();
    analisa_sebab['user_naik,call_naik,durasi_naik']= ' .....';
    analisa_sebab['user_naik,call_naik,durasi_turun']= ' .....';
    .....

    analisa_tindakan= Array();
    analisa_tindakan['user_naik,call_naik,durasi_naik']= ' .....';
    analisa_tindakan['user_naik,call_naik,durasi_turun']= ' .....';
    .....

    var j= analisa_sebab[kondisi].split(';');
    document.write('<br/>&lt;br/>&lt;b>Penyebab: &lt;/b>');
    for (i = 0; i &lt; j.length; i++)
        document.write("<br/>- "+j[i]);

    var j= analisa_tindakan[kondisi].split(';');
    document.write('&lt;br/>&lt;br/>&lt;b>Tindakan: &lt;/b>');
    for (i = 0; i &lt; j.length; i++)
        document.write("<br/>- "+j[i]);
}
```

Dan hasil yang didapat dari program tersebut adalah tampilan halaman analisa pertumbuhan seperti pada gambar berikut:

**Laporan Pertumbuhan Telkomnet@Home
Tahun 2004**

Bulan	Pengguna	Panggilan	Penggunaan
Juli	7	3	250
Juni	7	1	68
Mei	24	5067	11600506
April	52	14311	33390100
Maret	44	13021	31502935
Februari	21	10396	23724960

Analisa Pertumbuhan Bulan Juli

- Jumlah pertumbuhan pelanggan tetap
- Jumlah pertumbuhan panggilan naik 200%
- Jumlah pertumbuhan penggunaan naik 268%

Penyebab:

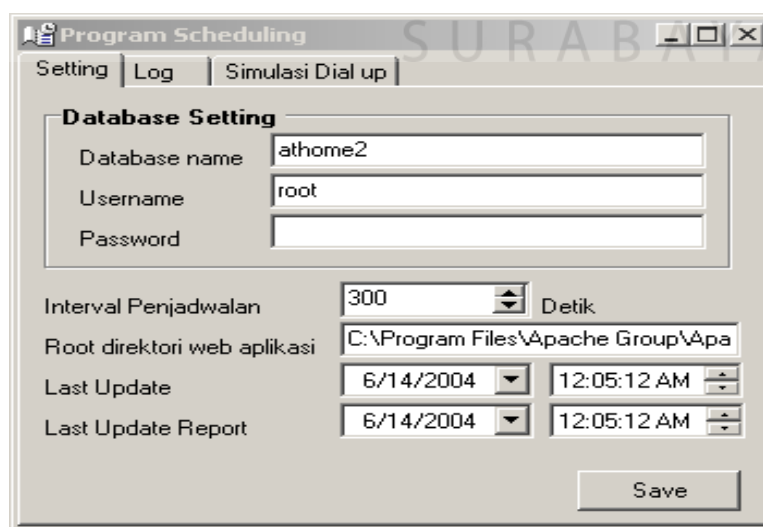
- Ada ketidakpuasan dari user yang out, User in lebih kecil dari user out
- Performansi jaringan dari yang out kurang baik

Gambar 4.15 Tampilan halaman analisa grafik pertumbuhan

4.2.3 Sistem aplikasi penjadwalan transformasi data

Penggunaan file XML pada sistem pendukung layanan TELKOMNet@Home memberikan beberapa peningkatan performa diantaranya dalam penyajian informasi yang cepat dan tepat, namun demikian ada beberapa konsekuensi yang tidak dapat dihindarkan, yaitu karakter sistem autentifikasi dan accounting dial-up (RADIUS) yang menggunakan media database sebagai sistem penyimpanan data. Hal ini tentu sedikit banyak berpengaruh pada implementasi konsep pemanfaatan teknologi XML yang dibuat pada Tugas akhir ini. Dalam hal ini adalah kebutuhan akan ketersediaan data file XML yang “siap saji”. Untuk itu diperlukan sebuah mekanisme untuk mentransformasi data tertentu yang secara default tersimpan pada tabel-tabel database MySQL menjadi file-file XML.

Sistem penjadwalan yang dibuat pada Tugas akhir ini ditujukan untuk melakukan proses transformasi data MySQL yang tersimpan pada tabel **RadAcct** dan tabel backup **Home[tahun][bulan]** menjadi file-file XML yang menyimpan data transaksi dial-up TELKOMNet@Home



Gambar 4.16 Tampilan program penjadwalan.

a. **Implementasi penjadwalan transformasi data MySQL menjadi file XML detil penggunaan**

Proses transformasi data dari database MySQL menjadi file XML tidak melibatkan semua data pada tabel, hanya data yang berkaitan dengan transaksi dial-up ke RADIUS dan data status pelanggan yang akan mengalami proses transformasi. Dengan demikian hanya tabel **RadAcct**, tabel **back-up** (home[tahun][bulan]) dan tabel **pelanggan** saja yang akan mengalami transformasi. Dari data pada ketiga tabel MySQL tersebut akan dihasilkan file XML detil penggunaan, file XML laporan tahunan dan update status pelanggan.

Berikut adalah potongan program yang memperlihatkan bagaimana proses transformasi data tabel RadAcct dilakukan.

Program 4.18 Program untuk memperlihatkan bagaimana proses transformasi data tabel RadAcct

```

procedure TForm1.updateFromRadacct;
var
  paket: TPaket;
begin
  //inisialisasi komponen MySqlQuery1
  MySQLQuery1.Server:= MySQLServer1;

  //Proses penulisan aktifitas yang akan dilakukan sekarang pada komponen Memo2 yang terdapat
  //pada tab Log
  Memo2.Lines.Add(FormatDateTime('dd\mm\yyyy hh:nn:ss',Now)+' : Start update detail usage
    from Radacct table');

  //Proses penulisan sintak SQL Select untuk memperoleh record pada tabel radacct yang
  memenuhi
  //syarat : AcctStopTime>= tanggal_terakhir_dilakukan_proses_transformasi
  MySQLQuery1.SQL.Clear;
  MySQLQuery1.SQL.Add('select *');
  MySQLQuery1.SQL.Add('from radacct');
  MySQLQuery1.SQL.Add('where acctstoptime>= ''+ FormatDateTime('yyyy-mm-dd hh:nn:ss',
    StrToDate(readStrConfig('lastupdate'))) +''');
  MySQLQuery1.SQL.Add('order by acctstoptime');
  MySQLQuery1.Open;

  //Untuk mendapatkan progres berupa indikator proses, ditambahkan komponen TProgressBar
  //berikut adalah proses inisialisasi property max pada komponen ProgressBar1
  ProgressBar1.Max:= MySQLQuery1.RecordCount;

  //Proses looping dlm mentransformasi setiap record baru tabel radacct yang belum
  //mengalami proses transformasi.
  while not MySQLQuery1.Eof do
  begin
    //prosedur standar agar selama proses transformasi yang lama, tidak menyebabkan program
  
```



```

//seolah-olah menjadi hang
Application.ProcessMessages;

//karena file xml detail penggunaan membutuhkan informasi biaya,jenis paket dsb. maka
//diperlukan prosedur yang menghasilkan output informasi paket dengan
//hanya membutuhkan parameter username dari field username tabel radacct
paket:= createPaket(MySQLQuery1.fieldbyname('username').AsString);

//Proses transformasi record tabel raddacct menjadi node pada file xml data penggunaan

updateDetail(MySQLQuery1.fieldbyname('radacctid').AsString,MySQLQuery1.fieldbyname('acctst
arttime').AsDateTime,
MySQLQuery1.fieldbyname('acctstoptime').AsDateTime,MySQLQuery1.fieldbyname('acctsessiont
ime').AsInteger,
MySQLQuery1.fieldbyname('username').AsString,paket);
//setelah sukses, dilakukan proses pencatatan record terakhir yang telah ditransformasi
setlastUpdate(MySQLQuery1.fieldbyname('acctstoptime').AsDateTime);

ProgressBar1.Position:= ProgressBar1.Position+1;
MySQLQuery1.Next;
end;

//Setelah proses selesai, semua komponen kembali diinisialisasi
ProgressBar1.Position:= 0;
Memo2.Lines.Add(FormatDateTime('dd\mm\yyyy hh:nn:ss',Now)+' : Finish update detail usage
from Radacct table');
end;

```

b. Implementasi penjadwalan transformasi data MySQL menjadi file XML report penggunaan per tahun

Aktifitas lain yang dijalankan selain transformasi data tabel radacct ke file XML detil penggunaan adalah pembuatan atau updating file XML report tahunan. Pada dasarnya file report tahunan merupakan resume dari angka-angka pertumbuhan, baik pertumbuhan pelanggan, penggunaan maupun durasi. Pada proses penjadwalan, hanya angka pertumbuhan penggunaan dan durasi saja yang mengalami transformasi. Sedangkan updating angka pertumbuhan pelanggan secara otomatis dilakukan pada saat customer care memasukkan data pelanggan baru melalui aplikasi web internal.

Untuk mekanisme transformasi data report tahunan sama dengan proses transformasi file XML detil pelanggan, yakni dengan cara melakukan query atas record-record tabel radacct yang memiliki nilai field acctstarttime lebih besar dari nilai terakhir field acctstarttime yang telah ditransformasi.

c. Implementasi simulasi dial-up

Proses dial up untuk menguji fungsional sistem dilakukan melalui simulasi. Adapun skenario yang digunakan untuk mewakili proses dial yang sesungguhnya dilakukan dengan cara memasukkan data penggunaan yang meliputi username, waktu mulai koneksi, waktu akhir koneksi, lama waktu koneksi dan sebagainya kedalam tabel RadAcct. Dengan demikian pada saat sistem aplikasi penjadwalan melakukan proses transformasinya sesuai dengan jadwal atau interval yang telah ditetapkan, akan terjadi penambahan data pada file XML detil penggunaan dan file XML laporan tahunan. Dan penambahan ini diharapkan dapat terlihat pada sistem aplikasi internal pada bagian laporan pertumbuhan jumlah panggilan dan penggunaan.

4.2.4 Sistem aplikasi konversi data pelanggan menjadi file XML data pelanggan

Pada bab tiga sebelumnya disebutkan bahwa sistem pendukung layanan TELKOMNet@Home melibatkan tiga jenis file XML, yakni file data pelanggan (cust.xml), file detil penggunaan (home[tahun][bulan]) dan file laporan pertahun (report[tahun].xml). Pada sistem aplikasi penjadwalan yang telah dijelaskan pada bagian sebelumnya, dapat disimpulkan bahwa dua dari tiga jenis file XML dihasilkan dari proses transformasi secara terjadwal, yakni file detil penggunaan dan file laporan per tahun. Adapun proses pembuatan atau updating file data pelanggan tidak disertakan ke dalam sistem aplikasi penjadwalan, dikarenakan masih ada kemungkinan untuk melakukan proses penambahan atau modifikasi data pelanggan secara real time, yaitu pada saat Customer Care melakukan penambahan data pelanggan melalui aplikasi web internal. Sistem konversi data

pelanggan yang dibuat memiliki dua fungsi utama, yaitu penambahan pelanggan dan update data pelanggan. Adapun proses transformasi yang dilakukan oleh aplikasi ini pada saat terjadi pemanggilan atas aplikasi ini dari halaman php. Sebagai contoh pada halaman insert.php yang akan dijalankan setiap kali Customer Care menambahkan pelanggan baru, dapat dijumpai baris program sebagai berikut “exec("BackAtHome.exe addpelanggan ".\$userhome);”, baris perintah tersebut berarti php memerintahkan aplikasi tersebut untuk melakukan penambahan node baru pada file XML data pelanggan, serta updating file XML report tahunan guna mengupdate angka pertumbuhan pelanggan dari yang sudah ada.

4.2.5 Sistem aplikasi rekening telepon (Intagjastel)

Salah satu pemanfaatan mekanisme penyimpanan data pada file XML antara lain dimungkinkannya proses sharing data. Pada sistem aplikasi rekening telepon (Intagjastel), proses sharing data dilakukan untuk mendapatkan informasi tagihan layanan TELKOMNet@Home. Secara garis besar, mekanisme pengambilan informasi tersebut dapat dilakukan dengan cara sebagai berikut:

- a. Melakukan pengecekan, apakah terdapat file detil penggunaan atas nomor telepon tertentu. Misal: untuk mengetahui apakah ada data penggunaan atas nomor 0313973185 untuk bulan April tahun 2004, maka perlu dilakukan pengecekan tentang keberadaan file dengan path “[rootdir]\2004\0313973185@home_200404.xml”.
- b. Jika tidak ditemukan, berarti tidak ada penggunaan layanan TELKOMNet@Home.

- c. Namun jika ada, proses berikutnya adalah mengambil informasi biaya total dan tambahan. Untuk kemudian digabung dengan biaya abonemen jasa non TELKOMNet@Home dan biaya *Jasnita* jika terdapat penambahan biaya. Namun untuk bulan pertama pendaftaran pelanggan hanya dikenakan tagihan biaya kelebihan permenit yang dikalikan dengan jumlah jam pemakaian, sedangkan untuk abonemen masih belum dibebankan.

Berikut adalah gambar dari aplikasi rekening telepon (Intagjastel):

The screenshot shows the 'Simulasi Intagjastel' application window. At the top left, there is a 'Telkomnet@Home' section with input fields for 'Username' (0318702568), 'Bulan' (Mei), and 'Tahun' (2004). Below these are two buttons: 'Ambil Data Telkomnet@Home' and 'Ambil Informasi Tagihan Telepon'. The main content area is titled 'INFORMASI TAGIHAN JASA TELEKOMUNIKASI' and features the Telkom Indonesia logo. It contains a table for 'Tgl. Akhir Bayar' (20/04/2004) and 'Bulan Tagihan' (April 2004). Below this, it says 'Kepada Yth,' followed by a box containing the customer's name and address: 'NY RUKMINI SUKMAWATI, JL. TUMAPEL No.06, KEPUTRAN, SURABAYA'. To the right of this box is another table with customer details: 'Nomor Pelanggan' (01389833), 'Nomor Jastel' (0315669692), 'Segmentasi' (Residensial), 'Grup' (A), and 'NPWP' (7805100127470). At the bottom, there are fields for 'Tagihan Bulan Ini (Rp.)' and 'Catatan Tagihan Bulan Lalu (Rp.)'. A watermark 'INSTITUT BISNIS & INFORMATIKA' is visible in the background.

Gambar 4.17 Tampilan aplikasi rekening telepon (Intagjastel).

4.2.6 Sistem aplikasi print out rincian penggunaan TELKOMNet@Home

Pada sistem aplikasi print out rincian penggunaan TELKOMNet@Home, proses yang dilakukan hampir sama dengan yang dilakukan pada sistem aplikasi rekening telepon (Intagjastel) sebagaimana diuraikan pada bagian sebelumnya. Proses diawali dengan melakukan pengecekan ada tidaknya file data penggunaan

dengan menggunakan parameter nomor telepon yang ingin dicek, beserta bulan dan tahun yang ingin dicek. Jika ditemukan, maka proses selanjutnya adalah menuliskan data penggunaan yang terdiri atas waktu mulai penggunaan, waktu selesai, lama penggunaan dan biaya yang dikeluarkan.

Berikut adalah gambar dari sistem aplikasi print out rincian penggunaan TELKOMNet@Home:

RINCIAN	NO TLP DIPANGGIL	TGL PERC	MULAI PERC	LAMA PERC (DETIK)	BIAYA

Gambar 4.18 Tampilan aplikasi print out rincian penggunaan TELKOMNet@Home.

4.3 Pengujian Sistem

Pada aplikasi pendukung layanan TELKOMNet@Home ini penulis melakukan testing pada program yang telah dibuat untuk dilakukan pengujian

apakah sistem yang dibuat bekerja dengan baik atau tidak, adapun testing yang dilakukan adalah sebagai berikut :

1. Testing fungsional dan tampilan halaman aplikasi

Pada testing ini dilakukan pengujian untuk seluruh halaman aplikasi dan form yang telah dibuat, masing–masing form diuji untuk mengetahui fungsi kerja masing–masing tombol apakah telah berjalan sesuai dengan desain atau tidak. Dari seluruh pengujian penulis mendapatkan hasil yang dapat dibagi menjadi 7 bagian sebagai berikut :

Tabel 4.1. Hasil pengujian fungsional dan tampilan sistem

No.	Pengujian	Hasil
1	Halaman Utama aplikasi internal	Tampil sesuai dengan desain, menu berfungsi dengan baik
2	Halaman Utama Administrator	Tampil sesuai dengan desain, menu berfungsi dengan baik
3	Halaman Utama Customer Care	Tampil sesuai dengan desain, menu berfungsi dengan baik
4	Halaman Utama Manajemen	Tampil sesuai dengan desain, menu berfungsi dengan baik
5	Masing – masing Form aplikasi internal	Tampil sesuai dengan desain, tombol dan validasi berfungsi dengan baik
6	Halaman Utama aplikasi eksternal	Tampil sesuai dengan desain, menu berfungsi dengan baik
7	Masing – masing Form aplikasi eksternal	Tampil sesuai dengan desain, tombol dan validasi berfungsi dengan baik

2. Testing kemudahan penggunaan

Testing ini bertujuan untuk mengetahui apakah sistem yang telah dibuat dapat dengan mudah digunakan oleh pengguna atau tidak. Untuk itu penulis telah meminta bantuan 10 orang dari internal TELKOM meliputi : Customer Care, Administrator, dan Manajemen, serta 10 orang dari pelanggan TELKOMNet@Home untuk mencoba dan memberikan penilaian terhadap aplikasi ini. Hasil yang diperoleh dari uji coba terhadap pengguna aplikasi adalah sebagai berikut (angket dapat di lihat di lampiran):

Tabel 4.2 Hasil angket uji coba.

Orang	Hasil Uji coba kepada Internal TELKOM	Hasil uji coba kepada Pelanggan
1	Mudah	Mudah
2	Mudah	Sedang
3	Sedang	Mudah
4	Sedang	Mudah
5	Mudah	Mudah
6	Sedang	Mudah
7	Mudah	Mudah
8	Sedang	Sedang
9	Mudah	Mudah
10	Mudah	Sedang

Dari hasil uji coba diatas didapatkan hasil bahwa, dari internal TELKOM: 60% menyatakan mudah, 40% menyatakan sedang, dan 0% menyatakan sulit. Sedangkan dari pihak pelanggan TELKOMNet@Home : 70% menyatakan mudah, 30% menyatakan sedang, 0% menyatakan sulit.

Dari hasil pengujian sistem, kesimpulan yang dapat ditarik penulis secara keseluruhan terhadap penggunaan program aplikasi tersebut diatas yang telah dilakukan pada internal TELKOM dan pelanggan TELKOMNet@Home menyatakan bahwa kecepatan akses dalam menampilkan data melalui query langsung ke XML lebih cepat dibanding query ke database MySQL, fungsional dan tampilan sistem telah berjalan dengan baik, sedangkan untuk pengoperasian juga cukup mudah untuk digunakan.

