

ARTIKEL ILMIAH TERPUBLIKASI

Email Autoresponder
Untuk Penyampaian Informasi Akademik
Di Fakultas Teknik Universitas Diponegoro

(Seminar Nasional Teknologi Informasi dan Aplikasinya, Politeknik Negeri Malang
Tgl. 12 Maret 2009, Malang, ISBN : 1979-6412, Hal : F-91 s/d F-96)



oleh :

Ir. Kodrat Iman Satoto, MT

**Program Studi Sistem Komputer
Fakultas Teknik
Universitas Diponegoro
2009**

MAIL AUTORESPONDER UNTUK PENYAMPAIAN INFORMASI AKADEMIK DI FAKULTAS TEKNIK UNIVERSITAS DIPONEGORO

Ir. Kodrat Iman Satoto, MT

ABSTRACT

As we all know that email is a means of communication for all Internet users an easy and efficient, so it can be used in all types of information systems including the Academic Information System. Academic information services using the email (email-based) that integrated with the Academic Information System can complete the method of delivery of information previously available using the web (web-based) and use the short message (sms-based).

In this research will be built a system of integrated mail autoresponder on the Academic Information System. This system works to provide academic information via email automatically. Autoresponder system is built with two scripts that filter email scripts to choose the order of information and a valid script to automatically using PHP.

Email-based academic information service is expected to be complete method of delivering information that has been previously available, namely the web-based and sms-based. Students and parents who want to get academic information can directly send the demand by sending an email with a certain format and will immediately get the information of interest automatically.

Keyword: web-based, SMS-based, email-based, Autoresponder

ABSTRAK

Seperti kita ketahui bersama bahwa email merupakan sarana komunikasi bagi seluruh pengguna Internet yang mudah dan efisien, sehingga dapat digunakan pada semua jenis sistem informasi termasuk Sistem Informasi Akademik (SIA). Layanan informasi akademik menggunakan email (email-based) yang terintegrasi dengan SIA dapat melengkapi metode penyampaian informasi yang sebelumnya sudah tersedia yaitu menggunakan web (web-based) dan menggunakan pesan singkat (sms-based).

Pada penelitian ini akan dibangun suatu sistem mail autoresponder yang diintegrasikan pada Sistem Informasi Akademik. Sistem ini berfungsi untuk memberikan layanan informasi akademik melalui email secara otomatis. Sistem autoresponder ini dibangun dengan dua skrip yaitu skrip filter email untuk memilih pesanan informasi yang valid dan skrip untuk menjawab secara otomatis menggunakan PHP.

Layanan informasi akademik berbasis email ini (email-based) diharapkan dapat melengkapi metode penyampaian informasi yang sudah tersedia sebelumnya, yaitu web-based dan sms-based. Mahasiswa dan orangtua/wali yang ingin mendapatkan informasi akademik dapat langsung mengirimkan permintaannya melalui pengiriman email dengan format tertentu dan akan segera mendapatkan informasi yang diinginkannya secara otomatis.

Keyword: web-based, SMS-based, email-based, autoresponder

PENDAHULUAN

LATAR BELAKANG

Saat ini email masih merupakan aplikasi yang paling banyak digunakan. Email adalah surat elektronik yang dikirimkan melalui jaringan komputer yang bersifat offline, artinya penerima tidak harus online saat menerima email. Sifat offline inilah yang malahan menjadi keunggulan dari email, karena pengirim dan penerima tidak harus online untuk mengirimkan pesan email. Keunggulan lainnya adalah tidak diperlukan bandwidth yang besar untuk pengiriman pesan email tanpa attachment.

Sistem informasi akademik yang sudah ada merupakan layanan Internet berbasis web. Sistem akademik ini berfungsi untuk memberikan informasi mengenai semua yang berkaitan dengan lembaga pendidikan yang dimilikinya khususnya informasi akademik. Informasi ditujukan baik untuk civitas akademik dalam lingkungan kampus maupun pemakai Internet secara umum.

Sarana penyampaian informasi menggunakan web mengharuskan pengguna selalu online, demikian juga menggunakan SMS hanya dapat mengirim pesan sebesar 160 karakter saja. Sedangkan email memungkinkan pengiriman pesan dalam jumlah besar, seperti transkrip akademik, tanpa harus selalu online.

Mahasiswa dan ortu/wali yang ingin mendapatkan informasi tertentu dari SIA dapat mengirimkan email dan menuliskan permintaan yang dimaksud dalam baris *subject*. Kemudian *mail-server* merespon permintaan yang ditulis di *subject* email dan mengirimkan informasi balasan yang dibutuhkan. Dalam layanan ini tidak perlu selalu terhubung dengan Internet jika ingin mendapatkannya, cukup mengirim permintaan dengan email dalam satu waktu, dan mengambil informasinya di lain waktu dengan hanya mengecek *mailbox*.

TUJUAN PENELITIAN

Tujuan dari penelitian ini adalah merancang sebuah program pada *mail-server* dimana *mail-server* yang telah ada pada web sistem informasi akademik dapat memberikan layanan informasi akademik secara *autoresponse* kepada mahasiswa. Permintaan yang dikirim oleh mahasiswa dan ortu/wali dalam bentuk email akan ditanggapi secara otomatis dan memberikan balasan informasi yang diminta.

KAJIAN PUSTAKA

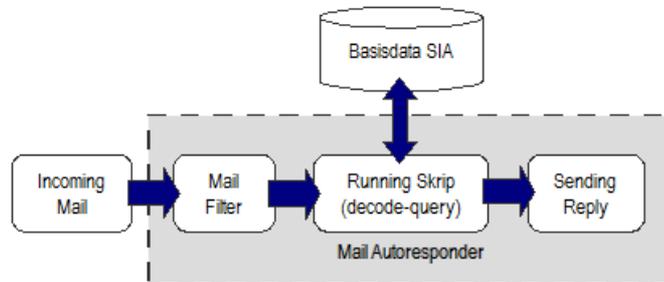
MAIL AUTORESPONSE

Mail autoresponse merupakan suatu sistem dimana dapat merespon secara otomatis email yang masuk ke suatu alamat email (*mail server*) dan kemudian melakukan *autoreply*. Autoresponder sudah banyak diaplikasikan dalam jaringan Internet, seperti yang dilakukan oleh Yahoo dan Google. Ketika kita mendaftar menjadi pengguna dalam Yahoo mail kita secara langsung mendapat email yang berisi sambutan dari Yahoo mail itu sendiri. Ini merupakan salah satu penerapan *mail autoresponse*.

Dalam mengirim email yang masuk bisa digunakan suatu *mail processing program* yang fungsinya menyaring email untuk menjalankan suatu skrip, salah satunya yaitu *Procmail*. *Procmail* adalah *mail processing program* yang ada di UNIX, dapat membantu menyelesaikan tugas-tugas berikut;

1. Menyaring email yang masuk (misalnya memisahkan pesan mailing-list dari pesan-pesan yang lain), sangat berguna untuk *preprocessing* email yang masuk dalam jumlah besar,
2. Mengurutkan email yang masuk berdasarkan kriteria, seperti pengirim, ukuran pesan atau keyword dalam kolom *subject*,
3. Mengimplementasikan mekanisme *autoreply* seperti membalas email kepada siapa

yang mengirim pesan secara otomatis.



Gambar 1.
Pemodelan Sistem *Mail Autoresponse*

Procmail dapat bekerja dengan cara membaca file yang dibuat dalam *home directory*, file ini bernama *.procmailrc*, file ini berisi instruksi untuk pemrosesan email yang masuk. Jika sebuah pesan cocok dengan kondisi yang ditulis, maka *procmail* mecegat pengiriman normal berdasar spesifikasi yang ditulis tadi. Sebagai contoh, pesan dapat diarahkan ke program atau diteruskan ke penerima yang lain, bahkan mungkin untuk menghasilkan *autoreply*. Jika tidak satupun kondisi yang cocok, maka *procmail* akan menaruh pesan ke *inbox default*.

Sebuah file *.promailrc* memiliki 2 bagian yaitu *assignments* dan *recipes*. Apa saja yang didahului dengan tanda (#) adalah komentar dan akan diabaikan. *Assignment* yang berada pada posisi atas berfungsi untuk men-setup variabel-variabel sehingga *procmail* tahu keberadaan program dan *mailbox*. *Recepies* merupakan penyaringan yang sesungguhnya berlangsung dan mempunyai format sebagai berikut;

```
:0 [flags] [: [lock-file] ]
zero or more conditions
one action line
```

Jika syarat (*condition*) terpenuhi, aksi (*action*) akan dilaksanakan, sehingga *procmail* dapat langsung digunakan untuk melakukan *mail autoreply* sederhana. Namun untuk sistem *mail autoreponse* yang melibatkan *query* data dan penyusunan informasi perlu ditambahkan skrip yang lain.

Suatu skrip dapat dijalankan dengan mengarahkan email masuk yang telah disaring sehingga email tadi menjadi *trigger* maupun input dari skrip yang telah dibuat. File *.procmailrc* ini melakukan penyeleksian berdasar *subject* pesan. *Subject* ditangkap untuk mengetahui permintaannya apa yang diinginkan pengirim. Selanjutnya email langsung diarahkan ke masing-masing folder dan skrip berdasarkan subject. Dalam penelitian ini skrip dibuat menggunakan bahasa pemrograman PHP untuk memproses permintaan berupa email.

SKRIP PHP

Permintaan email yang sudah dialihkan *procmail* ke skrip program PHP ditangkap sebagai input (stdin). Dalam mengolah permintaan program perlu men-*decode* pesan agar diperoleh *headers* dan *body*. Sudah tersedia suatu modul/paket dari PEAR (*PHP Extension and Application Repository*) yaitu *Mail_Mime* yang berguna untuk men-*decode* email sehingga dapat mempermudah proses identifikasi pesan.

Selanjutnya untuk membalas permintaan diperlukan fungsi *mail()* yang telah ada dalam PHP. PHP otomatis dapat memilih mengirimkan email menggunakan program eksternal seperti */usr/sbin/sendmail* yang default di Linux dan UNIX. Sintaks *mail()* yang digunakan untuk mengirim sebuah email adalah sebagai berikut;

```
mail($to, $subject, $message, [$additional_headers, [$additional_parameters]]);
```

`$additional_headers` adalah sebuah string yang terdiri dari satu atau lebih baris header tambahan yang ingin disebutkan. Biasanya yang paling sering disebutkan di sini adalah header *From* atau *Reply-To*. `$additional_parameters` digunakan untuk mengirim opsi ke program `/usr/sbin/sendmail` atau yang setara.

PEAR

PEAR (*PHP Extension and Application Repository*) merupakan repositori resmi dari php.net yang salah satu tujuannya untuk menyediakan library dan paket bagi pengguna PHP dengan kode-kode yang *open source*. Kelas-kelas dasar PEAR dan PEAR manager (`pear`) umumnya disertakan dalam distribusi standar PHP.

Dalam paket `Mail_Mime` terdapat beberapa kelas, salah satu kelas yang digunakan adalah `Mail_mimeDecode::decode()`. `Mail_mimeDecode` berisi fungsi-fungsi (*functions*) untuk membuat object `Mail_mimeDecode` baru, men-*decode* dan menghasilkan struktur yang berisi data pesan. Email yang diarahkan ke skrip menjadi input string bagi `Mail_mimeDecode` ini untuk kemudian di-*decode*, hasilnya berupa array misalnya `$structure`. Header-header disimpan di dalam `$structure['headers'] [<nama-header>]`, dimana `<nama-header>` adalah nama header yang telah dikonversi menjadi huruf kecil semua. Dengan demikian dapat diperoleh header pengirim (*From*), dan isi pesan (*Body*) untuk mengolah permintaan.

Header pengirim diambil untuk mendapatkan alamat email pengirim untuk membalas email (*reply*). Sedangkan *body* untuk keperluan *verifikasi* jika dibutuhkan, yaitu NIM dan Password. Dalam mengurai isi pesan dibutuhkan mekanisme *regex* (*regular expression*) sederhana. Hal ini berfungsi untuk pemisahan NIM dan Password sekaligus pencocokan formatnya, misalnya NIM harus memiliki format L2FXXXXXX (X = berupa angka tanpa spasi).

QUERY BASISDATA

Query basisdata digunakan untuk mengumpulkan data dari suatu basisdata untuk menghasilkan informasi yang dibutuhkan. Sebelum melakukan *query* basisdata maka koneksi dengan server basisdata harus dilakukan.

Tanpa melakukan koneksi dengan server basisdata maka tidak bisa untuk mengambil data tersebut (*query* basisdata). Untuk melakukan koneksi dengan server basisdata (MySQL), PHP menggunakan fungsi `mysql_connect()`, sintaks yang digunakan adalah;

```
mysql_connect(hostname, username, password);
```

Koneksi ke basisdata akan secara otomatis terputus ketika skrip program selesai dijalankan seluruhnya kecuali diberikan fungsi `mysql_close()`. Fungsi koneksi diatas akan memberikan nilai boolean TRUE apabila koneksi berhasil dan FALSE apabila gagal.

Apabila pengguna telah berhasil melakukan koneksi dengan server basisdata MySQL, pengguna dapat melakukan *query*. PHP tidak menyediakan fungsi khusus untuk melakukan operasi data, sehingga *query* yang digunakan adalah sintaks-sintaks dari MySQL. Tahapan untuk melakukan *query* data adalah sebagai berikut;

1. Koneksi ke basisdata,
2. Memilih basisdata yang akan diambil datanya,
3. Mendeklarasikan sebuah variabel string yang berisi sintaks perintah (*query*) MySQL yang akan dilakukan,
4. Melaksanakan sintaks MySQL menggunakan fungsi `mysql_query()`, jika sintaks yang akan dijalankan menghasilkan output yang akan ditampilkan, maka harus

- dideklarasikan sebuah variabel untuk menampung hasil tersebut,
- Mengambil hasil dari sintaks MySQL yang dilaksanakan menggunakan fungsi-fungsi seperti `mysql_fetch_array()`, `mysql_fetch_row()`, `mysql_num_rows()`, dan lain-lain tergantung pada format output yang diinginkan.

Sintaks yang digunakan untuk mengambil data dari basisdata tertentu adalah sebagai berikut:

```
SELECT field1,field2, ... FROM nama_table WHERE condition1, condition2, ...
ORDER BY field
```

Parameter ORDER BY untuk mengurutkan data berdasar *field*. Standar urutannya adalah dari angka terkecil atau dari huruf A dan dari data pertama masuk hingga data terakhir. Untuk mendapatkan urutan yang terbalik maka dibutuhkan atribut DESC.

METODOLOGI

Dalam melakukan analisis, digunakan beberapa metode yaitu studi literatur, pengujian sistem yang sudah ada, perumusan solusi permasalahan dan penerapan solusi permasalahan. Masing-masing metode memiliki keterkaitan satu dengan lainnya.

STUDI LITERATUR

Tahap ini dilakukan dengan mengumpulkan bahan-bahan pustaka (literatur) sesuai dengan masalah yang dihadapi. Literatur sebagian besar didapat dari Internet karena sebagian besar referensi buku cetak tidak ditemukan.

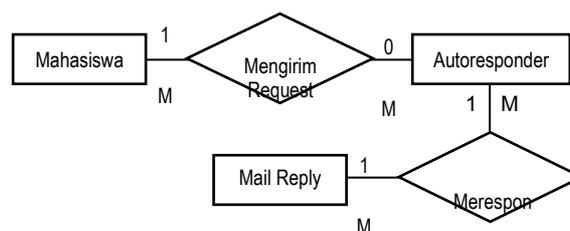
PERANCANGAN SISTEM

Perancangan sistem untuk layanan informasi akademik berbasis e-mail ini dilakukan dengan *Entity-Relationship Diagram* (E-R Diagram) dan *Data Flow Diagram* (DFD). Seperti sudah dijelaskan sebelumnya, sistem *autoresponse* ini akan menangkap *permintaan* dari mahasiswa dan ortu/wali yang berupa pesan e-mail, menyaring, men-decode, verifikasi, melakukan query basisdata dan mengirimkan informasi balasan sesuai permintaan.

E-R DIAGRAM

Diagram E-R merupakan suatu diagram yang digunakan untuk memodelkan hubungan antara entitas-entitas yang ada di lingkungan sistem. Tahapan rencana pendahuluan dalam program adalah menentukan lingkup sistem yang akan ditangani. Dengan menganalisa ide dasar diatas, maka dapat ditemukan sejumlah entitas untuk membentuk Diagram E-R, yaitu;

- Mahasiswa
- Sistem Autoresponse
- Mail Reply



Gambar 2. E-R Diagram Sistem *Mail Autoresponse*

IMPLEMENTASI DAN PENGUJIAN SISTEM

Implementasi dalam bentuk kode-kode program (*source code*) menggunakan bahasa pemrograman PHP. Dimana PHP disini adalah PHP dalam mode biner (*/usr/bin/php*) yang akan berjalan atas input dari *mail filter*, bukan mode modul Apache, karena tidak mengeksekusi skrip PHP via *browser*. Skrip PHP ditempatkan dalam *mail server* yang merupakan alamat tujuan user dalam mengirimkan *mail request*.

Pengujian terhadap sistem yang sudah dibangun adalah dengan mengirim email ke alamat yang sudah ditentukan dimana didalamnya terdapat *sistem autoresponse*. Email akan menjadi input bagi sistem, kemudian mengamati respon sistem hingga diperoleh kesesuaian antara *mail request* dengan balasan email dari sistem yang sesuai dari tujuan pembangunan sistem.

Sistem *mail autoresponse* ini digunakan untuk menangani enam *email request* yang berupa jadwal kuliah, jadwal ujian, jadwal dosen, kartu rencana studi (KRS), kartu hasil studi (KHS) dan transkrip nilai. Implementasi untuk pembuatan kode program PHP juga terdapat enam program, artinya satu program PHP hanya akan menangani satu *email request*, misalnya *email request* dengan subjek KRS akan ditangani oleh program PHP dengan nama file *krs.php*. Disamping pembuatan kode program PHP, juga yang tidak kalah penting adalah pembuatan skrip untuk *mail filter* (*.procmailrc*). Sebab dengan *mail filter* ini program PHP akan menerima *trigger* email sehingga program dapat berjalan.

KODE PROGRAM MAIL-FILTER

Pertama-tama yang dapat kita lakukan adalah membuat kode program *mail filter*. Dengan *procmail* kita dapat membangun *mail filter* dengan membuat file *.procmailrc* dan sebuah folder *procmail* yang berisi file *log*. Folder *procmail* ini ditempatkan pada direktori */home/dicky/*. File *.procmailrc* cukup ditaruh dalam direktori */home/dicky* sama dengan kode-kode program PHP lainnya yang akan dibuat. Berikut fungsi dari *Mail filter*;

1. Untuk meletakkan *mail request* yang masuk ke alamat email/mail server (dicky@localhost.localdomain) ke dalam folder yang sesuai subjek request dan
2. Mengumpulkan *mail request* ke program PHP sesuai dengan subjek requestnya. Artinya email tadi menjadi input dari program PHP sesuai isi subjeknya.

Skrip yang digunakan dalam file *.procmailrc* adalah sebagai berikut :

```
PATH=/usr/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin:/usr/games:/home/dicky/bin
SHELL=/bin/sh
MAILDIR=$HOME/Mail
DEFAULT=$HOME/Mail/inbox/
PMDIR=$HOME/.procmail
LOGFILE=$PMDIR/log
VERBOSE=yes
LOGABSTRACT=all
AUTOREPLYDIR=$HOME

SUBJEK=`formail -zxSubjek`

:0
*! ^X-Loop: dicky@localhost.localdomain
{
#Request KHS
:0c:
* ^Subjek:.*khs
khs/
```

```
:0
* ^Subjek:*khs
|$HOME/khs.php
```

Dengan skrip diatas email dengan subjek KHS yang masuk ke alamat dicky@localhost.localdomain akan ditaruh dalam folder /mailbox/khs dan selanjutnya dalam waktu yang berurutan mengumpankan email ke program khs.php. Namun jika email yang masuk tidak memiliki subjek yang berisi KHS maka akan diteruskan ke skrip berikutnya.

KODE PROGRAM PHP

Pada program PHP ini email akan diproses untuk mendapatkan informasi dan proses *autoreply*. Semua kode program PHP memiliki fungsi untuk mengolah *mail request* sesuai request yang diminta dan menyusun informasinya, kemudian mengirimkan balasan email tersebut. Setiap program PHP harus dapat menyelesaikan tugasnya untuk mendekode email hasil *filter* dari procmail. Pekerjaan ini dapat dilakukan dengan fungsi yang mengusung paket Mail_Mime, yang berfungsi untuk mendekode email sehingga bisa dipecah menurut *header–header* dan *body message*.

Semua kode program PHP mempunyai skrip untuk mendekode email dengan kelas Mail_mimeDecode yang masih merupakan bagian paket PEAR Mail_Mime. Skrip dimulai dengan membaca email hasil filter dari procmail yang diarahkan ke program PHP. Selanjutnya mendekode mail menggunakan kelas Mail_mimeDecode::(\$params), hasilnya disimpan dalam sebuah array dan apabila dilihat strukturnya sebagai berikut;

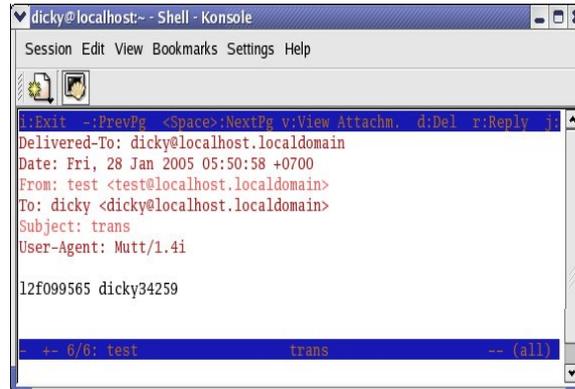
```
stdClass Object
(
    [headers] => Array
        (
            [received] => by localhost (Postfix, from userid 500) id 22F5920628; Wed, 19 Jan 2005 01:34:25 -0500 (EST)
            [date] => Wed, 19 Jan 2005 13:34:25 +0700
            [from] => mailuser <mailuser@localhost.localdomain>
            [to] => dicky@localhost.localhost
            [subjek] => khs
            [message-id] => <20050119063425.GA5532@localhost.localdomain>
            [mime-version] => 1.0
            [content-type] => text/plain; charset=us-ascii
            [content-disposition] => inline
            [user-agent] => Mutt/1.4i
        )
    [ctype_primary] => text
    [ctype_secondary] => plain
    [ctype_parameters] => Array
        (
            [charset] => us-ascii
        )
    [disposition] => inline
    [body] => L2F099565 dicky34259
)
```

Selanjutnya informasi yang ada di masing-masing *header* dan *body* dapat dengan mudah disimpan ke dalam sebuah variabel global. Informasi *body message* disimpan dalam variabel \$body, dan informasi alamat pengirim email disimpan dalam variabel \$sender.

Dalam proses verifikasi body message digunakan bantuan Reguler Expression (Regex), pencocokkan password dilakukan dengan enkripsi MD5. Sedangkan dalam melakukan proses pengiriman email kepada pengirim request digunakan fungsi mail().

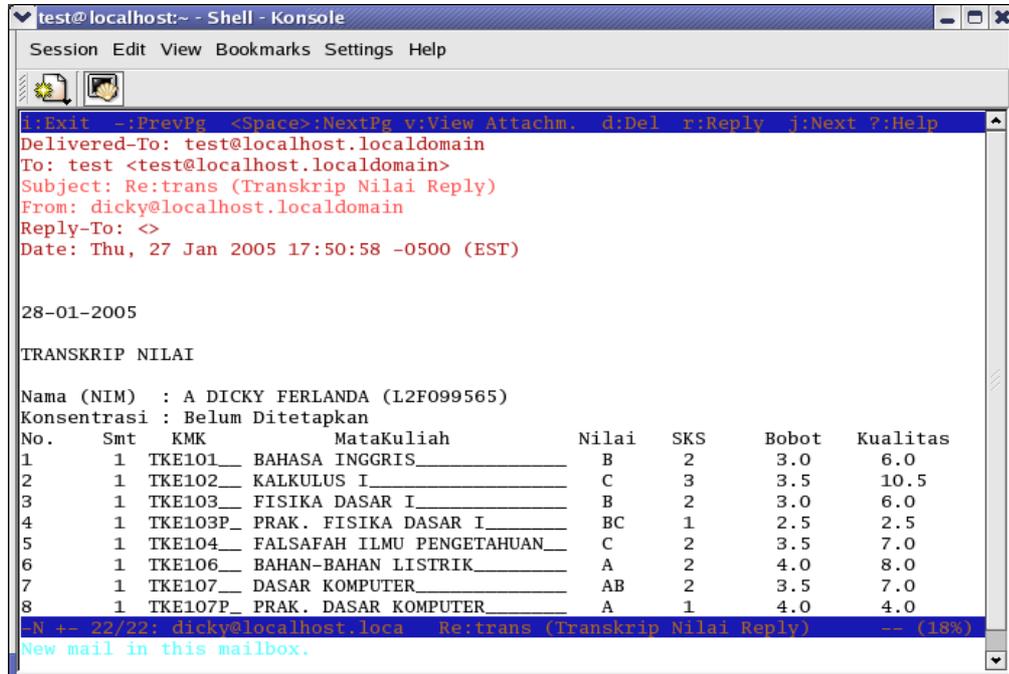
PENGUJIAN PERMINTAAN INFORMASI TRANSKRIP NILAI

Pengujian dimulai dengan mengirim email ke dicky, mengisi baris subjek dengan *trans*. Pada badan pesan harus diisi NIM dan password untuk keperluan verifikasi, dengan format NIM<spasi>password. Gambar 3 menunjukkan pengiriman email permintaan informasi transkrip nilai yang benar.



Gambar 3. Permintaan informasi transkrip nilai

Sistem akan merespon sesuai dengan permintaannya yang tertulis di subjek dan memberi balasan email seperti gambar 4 dan 5. Gambar 4 menunjukkan informasi transkrip nilai bagian atas, sedangkan gambar 5 menunjukkan informasi transkrip bagian bawah.



Gambar 4. Balasan informasi transkrip nilai (bagian atas)

Kedua gambar tersebut adalah satu email balasan dari *mail server*, yang diambil dari mailbox user. Tampak bahwa informasi transkrip nilai terdapat total SKS, bobot nilai sekaligus IPK.

Dalam *mail request* transkrip nilai juga mungkin terdapat kesalahan dalam menulis format NIM<spasi>Password. Jika hal ini terjadi maka sistem akan memberikan balasan email yang berisi pesan kesalahan, seperti yang terdapat dalam pengujian *mail request* KHS.

```

test@localhost:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help

1:Exit  -:PrevPg  <Space>:NextPg  v:View Attachm.  d:Del  r:Reply  j:Next  ?:Help
55      5  TKE377__  TEORI ANTREAN & KEANDALAN__  AB  2  3.5  7.0
56      6  TKE309__  PEMROGRAMAN C++__  B  3  3.0  9.0
57      6  TKE341__  PENGOLAHAN SINYAL DIGITAL__  AB  3  3.5  10.5
58      6  TKE381__  SISTEM BASIS DATA__  A  3  4.0  12.0
59      6  TKE382__  SISTEM FUZZY__  A  3  4.0  12.0
60      6  TKE383__  REKAYASA PERANGKAT LUNAK__  A  2  4.0  8.0
61      6  TKE383P__  PRAK. REKAYASA P. LUNAK__  A  1  4.0  4.0
62      6  TKE391E__  TEKNIK KOMPILASI__  AB  3  3.5  10.5
63      6  TKE463D__  JARINGAN KOMPUTER__  A  3  4.0  12.0
64      7  TKE307P__  PRAK. JARINGAN KOMPUTER__  AB  1  3.5  3.5
65      7  TKE311B__  PEMROGRAMAN JARINGAN__  D  3  1.0  3.0
66      7  TKE312B__  PEMROGRAMAN PLATFORM INDP__  A  2  4.0  8.0
67      7  TKE314B__  SISTEM INFOR. MANAJEMEN__  AB  3  3.5  10.5
68      8  TKE028__  KULIAH KERJA NYATA/MAGANG__  A  3  4.0  12.0
69      8  TKE031__  KULIAH KERJA LAPANGAN__  A  0  4.0  0.0

Total= 187
IPK = 2.9893048128342

+ 22/22 dicky@localhost.local Re:trans (Transkrip Nilai Reply) -- (end)

```

Gambar 5. Balasan informasi transkrip nilai (bagian bawah)

PENUTUP

KESIMPULAN

Dari pengujian *Sistem Autoresponse* diperoleh beberapa kesimpulan berikut;

1. Bahasa pemrograman yang dipakai dalam membangun sistem adalah PHP dengan *mode biner* karena program berjalan atas input dari email *request* bukan dari *request* dari *browser client*,
2. Email *request* yang merupakan input bagi skrip PHP adalah email yang telah terfilter oleh *procmail* berdasarkan subjeknya,
3. *Sistem autoresponse* dapat diterapkan dalam Layanan Informasi akademik via email. Sedangkan layanan yang dapat ditangani berupa *mail request* KHS, KRS, jadwal kuliah, jadwal dosen, jadwal ujian dan transkrip nilai,
4. Untuk mendapatkan informasi *headers* dan *body* dari email, program PHP menggunakan sebuah paket dari PEAR yaitu *Mail_Mime*, yang berfungsi untuk mendekode email,
5. *Sistem autoresponse* juga dapat mendeteksi adanya kesalahan dalam penulisan dalam *body message* dan mengirimkan balasan email berupa pesan kesalahan yang terjadi.

SARAN

Setelah dilakukan pengujian pada *sistem autoresponse* yang telah diimplementasikan untuk layanan informasi akademik, dapat diberikan beberapa saran sebagai berikut;

1. Layanan dapat dikembangkan menjadi email yang dapat menginput ke sistem informasi akademik (basidatanya) tidak hanya berupa email request, misalnya dapat

- digunakan untuk entri data krs melalui email,
2. Informasi dalam email balasan masih berupa *text-mode*, selanjutnya dapat dikembangkan *email attachment* yang berisi data yang *comaptible* dengan MS-Excel, sehingga akan memudahkan mahasiswa dalam analisis informasi akademik.

DAFTAR PUSTAKA

1. Dennis, Jim, "*Procmail Mini-Tutorial: Automated Mail Handling*", <http://www.linuxgazette.com/issue14/procmail.html>
2. Hakim, Lukmanul dan Musalini, Uus, "*150 Rahasia dan Trik Menguasai PHP*", PT Elex Media Komputindo, Jakarta 2003.
3. Haryanto, Steven, "*PHP: Kumpulan Resep Pemrograman* ", PT Dian Rakyat, Jakarta, 2004.
4. Haryanto, Steven, "*Regex: Kumpulan Resep Pemrograman* ", PT Dian Rakyat, Jakarta, 2004.
5. Plew, Ronald R. and Stephens, Ryan K., "*SAMS Teach Yourself SQL in 24 Hours*", Sams Publishing, Indianapolis – Indiana, USA, 2000.
6. Salmi, Timo, "*Timo's procmail tips and recipes*", <http://www.uwasa.fi/~ts/info/proctips.html>, 2003.
7. Soboroff, Ian, "*Mail Filtering with Procmail* ", <http://userpages.umbc.edu/~ian/procmail.html>, 1997
8. The PEAR Documentation Group, "*PEAR Manual*", 2004.
9. The PHP Documentation Group, "*PHP Manual*", 2004.
10. Wankyu Choi, Allan Kent, Chris Lea, Ganesh Prasad, Chris Ullman, "*Beginning PHP4*", Wrox Press Ltd, Birmingham, UK, 2000.
11. Welling, Luke and Thomson, Laura, "*PHP dan MySQL Web Development*", Sams Publishing, Indianapolis – Indiana, USA, 2003.
12. -----, "*Procmail FAQ*", <http://rhols66.adsl.netsonic.fi/era/procmail/mini-faq.html>
13. -----, "*Procmail*", <http://www.udel.edu/topics/e-mail/procmail/procmail.html>