

**PENGEMBANGAN SISTEM PEMROSESAN
TRANSAKSI ADMINISTRASI AKADEMIK PERGURUAN TINGGI
MELALUI INTERNET**

Oleh:
Samsul Hadi
Universitas negeri Yogyakarta

Abstract

The purpose of this research was to developed a web-based transaction processing system of higher education academic administration that considered data security. The system consisted of tree parts of software, one for network administrator, one for academic administrator staff, and the other for student. The function of each part should consider the user authorization. This research used Windows 97 operating system. Microsoft Access 97, WebBase 4.10 build 57, and a computer supported the software that was ready to be connected to computer network. Thirteen tables were built to store data. The system software was written in HTML, WebBase macro, and SQL to modify data in the tables. The software was saved in htf files that could be accessed with browser. The research found that the expected system could function well.

key words: webbase, internet, transaction processing system.

Pendahuluan

Jaringan komputer global atau internet berkembang dengan pesat, dalam 10 tahun memiliki \pm 40 juta pelanggan (Brian, dkk, 1994). Di Indonesia, internet juga sudah banyak digunakan. Saat ini telah banyak penyedia jasa internet atau ISP (Internet Service Provider), Bahkan ada ISP yang telah mempunyai cabang di seluruh propinsi di Indonesia.

Di sisi lain, wilayah kerja suatu perguruan tinggi secara geografis tidak dibatasi, Ada perguruan tinggi yang mempunyai kampus terpisah-pisah dengan jarak yang relatif jauh. Selain itu,

lulusan SLTA dari manapun di wilayah Indonesia, bahkan dari luar negeri, dapat mendaftar di perguruan tinggi yang mereka inginkan. Saat libur mahasiswa biasanya pulang ke asalnya dan kembali ke kampus untuk menyelesaikan administrasi akademik dengan membayar SPP, mengisi KRS, dan mengurus nilai melalui KHS atau transkrip, serta mengikuti kuliah.

Selain melibatkan mahasiswa, administrasi akademik melibatkan petugas. Petugas menangani pendaftaran mahasiswa, pengumpulan KRS, pembuatan KHS dan transkrip. Kegiatan tersebut terkait dengan data program studi, fakultas, institut/universitas, dosen, waktu pendaftaran, daftar mata kuliah, nilai mata kuliah, jumlah peserta mata kuliah, serta data yang berkaitan dengan peraturan akademik misalnya SKS maksimum yang dapat diambil berdasarkan perolehan IP (indeks prestasi) mahasiswa.

Dalam kaitannya dengan usaha peningkatan ragam pemanfaatan internet, kegiatan administrasi akademik perlu dirintis untuk dilakukan melalui media ini. Untuk itu perlu dibuat perangkat lunak yang menunjang. Perangkat lunak yang dibuat harus mempertimbangkan keamanan data. Tidak semua orang dapat memanfaatkan fasilitas tersebut kecuali telah diberi kewenangan. Mahasiswa perlu diberi kewenangan oleh petugas setelah ia membayar SPP atau telah memenuhi persyaratan tertentu. Namun petugas sendiri juga harus mendapat kewenangan dari administrator jaringan untuk memanfaatkan fasilitas tersebut.

Sistem Pemrosesan Transaksi (SPT)

SPT atau dalam bahasa Inggrisnya TPS (*transcation processing system*) adalah salah satu jenis sistem informasi. Transaksi adalah kegiatan yang dapat membangkitkan atau memodifikasi data (Alter, 1992). Transaksi juga dapat diartikan sebagai

satuan kegiatan yang berisi satu atau lebih perintah pemrosesan data (Whalen dan DeLuca, 1998). Dengan demikian maka SPT dibuat untuk mengumpulkan dan menyimpan data serta mengendalikan keputusan yang terkait dengan kegiatan transaksi.

Cara pemrosesan transaksi adak dua jenis, yaitu *batch processing* dan *realtime processing*. Dalam *batch processing*, data transaksi individual dikumpulkan dan disimpan terlebih dahulu, data tersebut tidak segera diproses. Bila waktunya telah sesuai jadwal atau jumlah data telah dianggap cukup, data baru diproses. Dalam *real time processing*, data transaksi diproses segera setelah transaksi dilakukan tidak perlu menunggu waktu dan jumlah data transaksi.

Salah satu cara untuk melakukan transaksi *realtime* adalah melalui internet. Internet menggunakan arsitektur *client/server* (Savola, dkk. 1996). Program *client* yang dikenal sebagai *browser* bekerja di komputer pengguna untuk mengirimkan permohonan (*request*) ke program *server* yang bekerja di komputer *server*. Baik komputer yang berfungsi sebagai *cliet* maupun *server*, bila terhubung ke internet harus memiliki alamat IP (*internet protocol*). Alamat ini bersifat unik, tidak ada dua komputer yang memiliki alamat IP yang sama (Tanenbaum dan Priatna, 1997).

Bahasa yang mula-mula digunakan untuk berkomunikasi antara *browser* dengan *server* adalah HTML (*Hypertext Markup Language*). HTML pertama kali dikembangkan di Laboratorium Fisika Partikel CERN, Swiss, tahun 1989. HTML terbaru adalah versi 3.2 yang diumumkan oleh konsorsium WWW pada bulan Mei 1996 (Ladd, dkk., 1996). Elemen HTML cukup banyak sehingga dapat menampilkan teks, gambar, dan suara. Selain itu HTML dapat digunakan untuk membuat form guna mengirimkan data dari *browser* ke *server* (Ladd, dkk. 1996).

WebBase adalah salah satu program server yang dapat menangani basis data (ExpertTelligence, Inc., 1997). WebBase mempunyai variabel dinamik yang dapat digunakan untuk mendeteksi alamat *server*, alamat *browser* dan waktu. Dengan program *server* ini maka dapat digabung informasi yang berasal dari basis data dengan informasi yang dibuat HTML. Agar tujuan tersebut dapat dilakukan, maka diperlukan perangkat lunak yang ditulis dengan HTML. *macro WebBase*, dan SQL (*Structured Query Language*).

Macro WebBase dipakai untuk menyisipkan file atau bagian program, membuat variabel baru, menampilkan variabel dalam format tertentu, melakukan pencabangan, iterasi, proteksi kesalahan, *query* berbasis data, menampilkan hasil *query* ke *browser*, serta melakukan operasi matematika, logika, *string*, dan waktu.

SQL merupakan bahasa yang digunakan untuk mengakses basis data. Hampir semua bahasa pemrograman dan antarmuka basis data menampung SQL standar yang telah ditentukan oleh komite ANSI-SQL (*American National Standards Institute - SQL*). SQL standar yang pertama kali digunakan adalah SQL-89. Standar ini diperkenalkan tahun 1989. SQL dapat digunakan oleh pemrogram komputer untuk memodifikasi basis data, membuat sistem keamanan basis data, dan membaca basis data untuk mendapatkan informasi (Stephen, 1997).

Model basis data yang paling populer adalah basis data relasional (Stephens, dkk. 1997). Basis data relasional adalah sekumpulan data yang saling berhubungan (Amundsen dan Smith, 1996). Basis data sendiri mempunyai arti sekumpulan data yang disimpan sedemikian rupa sehingga mudah diakses (Whalen dan DeLuca, 1998). Basis data dapat berisi banyak tabel. tabel-tabel yang ada dapat saling berhubungan.

Bila basis data dipublikasikan di internet, maka akan dapat dibaca orang melalui komputer lain yang sama-sama tersambung ke internet. Basis data tersebut perlu diamankan dari maksud-maksud yang tidak baik. Terdapat banyak cara untuk mengamankan data yang dipublikasikan di internet. Penggunaan *account*, *password*, *firewall*, dan penempatan skrip atau program di direktori yang tidak diketahui pihak luar merupakan cara yang biasa dilakukan (Morgan, dkk. 1996, ExperTelligence, Inc. 1997, Huches Technologies, 1997, dan Savola, dkk. 1996).

Cara Penelitian

Alat dan Bahan Penelitian menggunakan komputer dengan *processor* Pentium MMX 200 Mhz., *harddisk* 3,2 GB, *diskdrive* 1,4 MB, *keyboard*, *mouse*, kartu VGA 1 MB, RAM 32 MB kartu Ethernet, dan monitor SVGA berwarna yang siap disambung ke jaringan. Perangkat lunak yang digunakan adalah sistem operasi Windows 97, WebBase 4.10 build 57, dan Microsoft Access 97.

Perangkat lunak yang dikembangkan dalam penelitian ini bersifat umum, karena itu harus bisa di *setup* atau disesuaikan dengan kondisi perguruan tinggi yang menggunakan. perangkat lunak ini diharapkan juga dapat mengenali petugas. Agar perangkat lunak dapat berfungsi seperti yang diharapkan, maka dibuat tabel basis data yang dapat menampung data hasil *setup* dan data petugas. Salain itu, perangkat lunak yang dikembangkan dalam penelitian ini diharapkan dapat menangani data yang berkaitan dengan administrasi akademik.

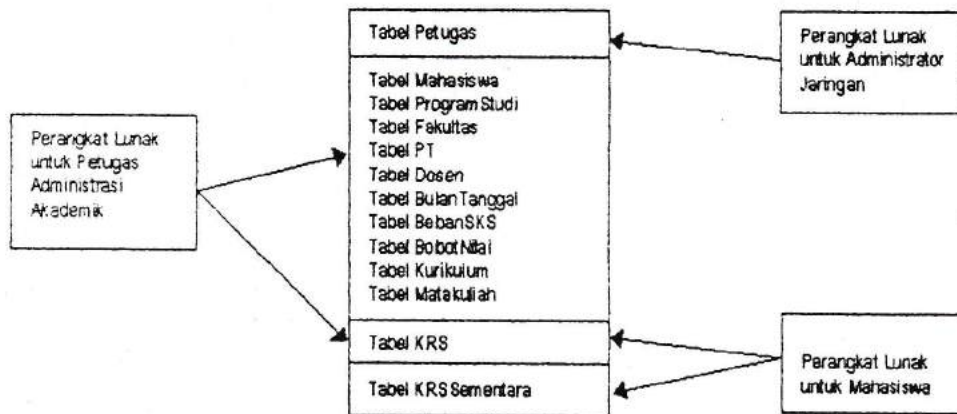
Tabel basis data yang digunakan untuk menampung data hasil *setup* dan data petugas adalah bersifat independen, yaitu tidak mempunyai relasi dengan tabel lain. Tabel yang bersifat independen

ini terdiri dari tabel PT, beban SKS, bobot nilai, bulan, tanggal, dan petugas.

Keempat tabel yang disebut terdahulu telah diberi nilai awal (*default*) pada tiap *field*-nya. Manipulasi data yang dilakukan terhadap keempat tabel tersebut adalah sekedar meng*update* nilai awal yang ada, tidak untuk diisi data baru atau dihapus, tabel petugas dapat diisi, di*update*, dihapus, atau sekedar dilihat isinya. Tabel petugas juga diberi nilai awal. Dengan nilai awal ini, administrator jaringan dapat menggunakan perangkat lunak yang dibuat untuknya guna memberi kewenangan kepada petugas administrasi akademik. Nilai awal tersebut kemudian dapat diubah untuk menjaga kerahasiannya.

Tabel basis data yang digunakan untuk menampung data administrasi akademik adalah bersifat relasional. Pembuatan tabel relasional menggunakan teknik *entry relationship*. Tabel ini terdiri dari tabel Mahasiswa, Kurikulum, Matakuliah, ProgramStudi, Fakultas, Dosen KRS, dan KRSSementara dengan *field* yang terkait dengan data administrasi akademik. tabel tersebut dapat diisi dengan data baru dan data yang sudah ada dapat dilihat, di*update*, atau dihapus.

Setelah pembuatan tabel, kemudian dibuat perangkat lunak untuk memodifikasi isi tabel. Perangkat lunak tersebut dibuat dengan bahasa HTML, makro WebBase, dan SQL menggunakan teks editor Notepad. Perangkat lunak ini disimpan dalam *file* dengan ekstensi htf. Perangkat lunak yang ada terdiri dari tiga bagian, masing-masing adalah untuk administrator jaringan, petugas administrasi akademik, dan mahasiswa. Fungsi ketiga perangkat lunak tersebut ditunjukkan dalam Gambar 1 berikut.



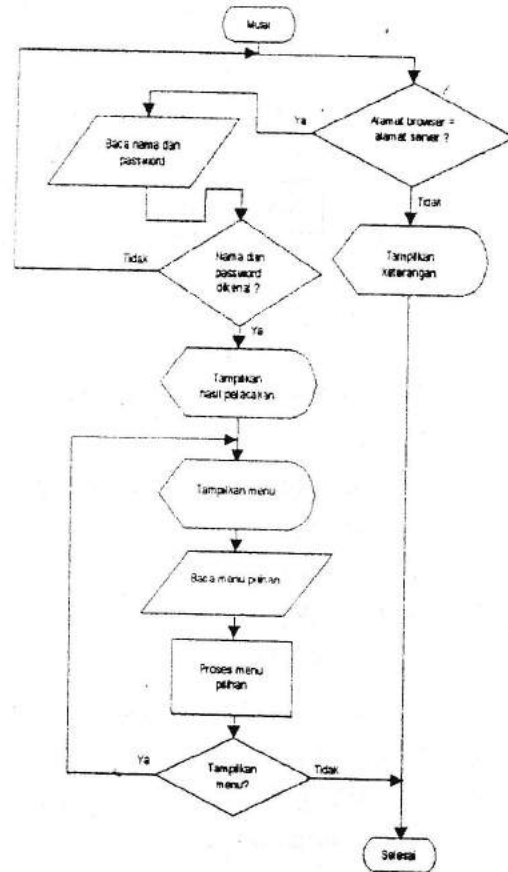
Gambar 1.
Fungsi Perangkat Lunak yang Ada

Perangkat lunak untuk adminisitrator jaringan digunakan untuk memodifikasi data dalam tabel Petugas. Perangkat lunak untuk petugas administrdsi akademik digunakan untuk memodifikasi data dalam tabel Mahasiswa, ProgramStudi, Fakultas, PT, Dosen, KRS, BulanTanggal, BebanSKS, BobotNilai, Kurikulum, dan Matakuliah. Perangkat lunak untuk mahasiswa digunakan untuk memodifikasi data dalam tabel KRS dan KRSSementara.

Diagram alir perangkat lunak ini secara garis besar ditunjukkan pada Gambar 2. Tampilan awal perangkat lunak ini menguji apakah alamat *browser* atau alamat IP komputer pengguna sama dengan alamat *server*. Jika alamat *browser* tidak sama dengan alamat *server*, maka ditampilkan keterangan bahwa alamat yang dituju oleh

browser tidak dapat dihubungi. Pengujian alamat browser dan alamat server menggunakan perintah sebagai berikut:

```
{if %serverAddress~ %browserAddress% = not}  
  <BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><CENTER>  
  Alamat tersebut tidak dapat dihubungi.  
</CENTER>  
{exit}  
{/if}
```



Gambar 2.
Diagram Alur Perangkat Lunak untuk Adminsitator Jaringan

dan

Jika alamat *browser* sama dengan alamat *server*, maka akan muncul tampilan yang meminta nama dan *password* pengguna. Nama dan *password* yang dimasukkan pengguna kemudian dibandingkan dengan data yang ada dalam tabel Petugas. Pengujian nama dan *password* yang dirancang dalam penelitian ini dapat membedakan huruf besar dan kecil (*case sensitive*). Nama dan *password* dianggap valid jika data yang dimasukkan oleh pengguna sama persis dengan data yang tersimpan dalam tabel, termasuk besar kecilnya huruf.

Pengujian nama password tersebut dilakukan menggunakan perintah sebagai berikut:

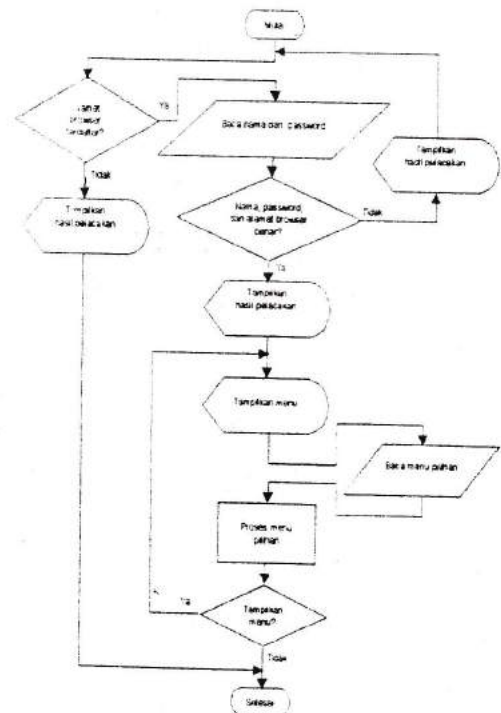
```
{sql to ptg ...}
  SELECT * FROM Petugas WHERE NamaPetugas=
    '{Nama}' and PINPetugas='{PIN}'
{/sql}
...
{set x NamaPetugas Nama=}
{set y PINPetugas PIN =}
{set z x y .&}
{if false z =}
  . . .
{else}
<H3>Hasil Pelacakan:</H3>
  . . .
  Anda terdaftar sebagai administrator
{/if}
```

Dalam perintah tersebut, variabel Nama yang diperoleh dari *form* dibandingkan dengan variabel *field* NamaPetugas dan hasilnya disimpan dalam variabel x. Variabel PIN yang diperoleh dari *form* dibandingkan dengan variabel *field* PINPetugas dan hasilnya disimpan dalam variabel y. Variabel x dan y kemudian dikombi-

nasikan dengan operator AND (&) dan hasilnya disimpan dalam variabel z. Variabel x, y, dan z bisa bernilai benar atau salah.

Bila nama dan *password* valid, yang dinyatakan dengan z bernilai benar, maka muncul tampilan yang menyatakan bahwa pengguna terdaftar sebagai administrator dan ia dapat menggunakan menu yang tersedia. Menu perangkat lunak ini adalah 1) Isi Petugas, 2) Lihat/*Update* Petugas, dan 3) Hapus Petugas. Jika nama dan *password* tidak valid, yang dinyatakan dengan z bernilai salah, maka akan muncul kembali tampilan yang meminta nama dan *password*.

Perangkat lunak untuk petugas administrasi akademik. dalam diagram alir secara garis besar ditunjukkan dalam Gambar 3.



Gambar 3.
Diagram Alir Perangkat Lunak
untuk Petugas Administrasi Akademik

Tampilan awal perangkat lunak ini menguji apakah alamat *browser*, yang dideteksi secara otomatis oleh *server*, terdapat dalam tabel Petugas atau tidak. Jika alamat *browser* tidak terdapat dalam tabel Petugas, maka akan ditampilkan keterangan bahwa alamat yang dituju tidak dapat dihubungi dan alamat *browser*. Proses tersebut dilakukan dengan perintah sebagai berikut:

```
{sql to awal ...}
  SELECT AlamatIP FROM Petugas Where AlamatIP
    = '{%browserAddress%}'
{/sql}
{if 0 awal size =}
  <BR><BR><BR><BR><BR><BR><BR><BR><BR><BR>
  <CENTER>Alamat tersebut tidak dapat
  dihubungi<BR>
  dari Alamat IP: {%browserAddress%}</CENTER>
  {exit}
{/if}
```

Jika alamat *browser* terdapat dalam tabel, maka akan muncul tampilan yang menanyakan nama dan *password* pengguna. Nama dan *password* yang dimasukkan pengguna, bersama dengan alamat IP komputer atau alamat *browser* yang dideteksi secara otomatis oleh *server*, kemudian dibandingkan dengan data yang ada dalam tabel Petugas.

Pengujian nama dan *password* di sini juga *case sensitive*. Jika nama, *password*, dan alamat *browser* tidak sesuai dengan data yang ada dalam tabel, maka muncul pemberitahuan bahwa pengguna tidak terdaftar sebagai petugas. Jika nama, *password*, dan alamat *browser* sesuai dengan data yang ada dalam tabel, maka akan ditampilkan pemberitahuan bahwa pengguna tercatat sebagai petugas, dan dapat

menggunakan menu yang ada berulang kali. Pengujian nama, *password*, dan alamat *browser* tersebut caranya mirip dengan pengujian nama dan *password* pada perangkat lunak untuk administrator jaringan.

Menu perangkat lunak ini ada 7 buah, masing-masing adalah: 1) Data PT, 2) Waktu Pendaftaran, 3) Data Kurikulum/Matakuliah, 4) Data Dosen, 5) SKS Maksimum, 6) Bobot Nilai, dan 7) Data Mahasiswa. Menu Waktu Pendaftaran, SKS Maksimum, dan Bobot Nilai tidak mempunyai submenu. Menu lainnya mempunyai submenu bahkan sub-submenu. Menu, submenu, dan sub-submenu tersebut digunakan untuk memodifikasi data administrasi akademik.

Perangkat Lunak untuk Mahasiswa dalam garis besar bagian ini ditunjukkan pada Gambar 4. Tampilan awal perangkat lunak ini juga meminta nama dan *password* pengguna. Nama dan *password* tersebut dibandingkan dengan data yang ada dalam tabel Mahasiswa. Pengujian nama dan *password* di sini juga *case sensitive*. Jika nama dan *password* tersebut ada dalam tabel, maka ditampilkan pemberitahuan bahwa pengguna terdaftar sebagai mahasiswa dan dapat menggunakan menu yang ada, jika tidak maka akan muncul pemberitahuan bahwa pengguna tidak terdaftar sebagai mahasiswa.

Menu yang ada ditentukan oleh waktu yang ada di komputer *server* serta pernah tidaknya mahasiswa mengisi KRS. Jika waktu menunjukkan saat pendaftaran semester ganjil dan pengguna belum mengisi KRS untuk periode tersebut, maka yang keluar adalah menu semester ganjil yang dilengkapi dengan tahun ajaran serta tanggal/bulan/tahun. Menu ini berisi submenu: 1) Isi KRS, 2) Lihat KRS, dan 3) Lihat Transkrip. Untuk semester genap kejadiannya sama, bedanya adalah menu yang keluar merupakan menu semester genap.

Jika waktu *server* menunjukkan masa pendaftaran semester ganjil dan pengguna sudah mengisi KRS untuk periode tersebut, maka menu yang keluar adalah menu antar-semester yang dilengkapi dengan tanggal/bulan/tahun. Menu ini berisi submenu: 1) Lihat KHS, dan 2) Lihat Transkrip dengan tambahan keterangan bahwa mahasiswa telah mendaftar semester ganjil pada tanggal/bulan/tahun seperti yang ada dalam basis data. Untuk semester genap kejadiannya sama. Pengujian waktu dan penampilan menu yang sesuai dilakukan dengan perintah sebagai berikut:

```
{sql to bulan ... }
  SELECT * FROM BulanTanggal
{/sql}
{set Waktu %dateTime%}
{forRow x on bulan}
  {set AwlGanjil AwalGanjil asUniversalTime}
  {set AkhGanjil AkhirGanjil asUniversalTime}
  {set AwlGenap AwalGenap asUniversalTime}
  {set AkhGenap AkhirGenap asUniversalTime}
{/forRow}
{set Wkt1 AwlGanjil AkhGanjil Waktu between:and:}
{set Wkt2 AwlGenap AkhGenap Waktu between:and:}
{set Wkt3 Wkt1 Wkt2 |}
{set Wkt4 Wkt3 r,ot}
{if true Wkt1 =}
  {insert 'menul.htf'}
{/if}
{if true Wkt2 =}
  {insert: 'menu2.htf'}
{/if}
{if true Wkt4 =}
  {insert 'menuatr.htf'}
{/if}
```


Perintah tersebut akan membaca data dalam tabel Bulan-Tanggal, dilanjutkan dengan membuat variabel Waktu yang nilainya sama dengan waktu yang ada di server menggunakan variabel dinamik %dateTime%. Data hasil pembacaan tabel BulanTanggal diurai dengan pernyataan {forRow... } dan {/forRow} untuk membuat variabel AwlGanjil, AkhGanjil, AwlGenap, dan AkhGenap yang nilainya sama dengan nilai variabel *field* yang ada, tetapi formatnya disamakan dengan format variabel Waktu. Variabel Waktu diuji dan hasilnya disimpan dalam variabel Wkt1, Wkt2, dan Wkt4 untuk menyatakan waktu semester ganjil, semester genap, dan antar-semester. Berdasarkan hasil pengujian Waktu, kemudian ditampilkan menu yang sesuai.

Hasil Penelitian dan Pembahasan

Hasil pengujian menunjukkan bahwa perangkat lunak yang dibuat dapat berfungsi dengan baik. Hal ini membuktikan bahwa HTML dapat digunakan untuk menampilkan informasi dalam berbagai format dan membuat form untuk menangkap data melalui *browser* dan mengirimkannya ke server (Ladd, dkk, 1996; Savola dkk., 1996; dan Kurniawan, 1997).

Hasil penelitian ini juga membuktikan bahwa makro WebBase dapat melakukan fungsi iterasi dan pencabangan, serta dapat digunakan untuk memanipulasi data numerik, teks, dan waktu (ExperTelligence, 1997). Selain itu, hasil penelitian ini juga sejalan dengan apa yang dikemukakan oleh Stephens dkk (1997), bahwa SQL dapat digunakan oleh pemrogram komputer untuk memodifikasi basis data, membuat sistem keamanan basis data, dan membaca basis data untuk mendapatkan informasi.

Pengujian keabsahan pengguna yang dikembangkan dalam penelitian ini menggunakan *form HTML*, tidak menggunakan *login dialog box*-nya *browser*. Penggunaan *form* tersebut dapat lebih komunikatif, karena format dan bahasa tampilannya bisa disesuaikan dengan pengguna, selain itu juga memungkinkan untuk diakses dengan sembarang *browser* (Walter, 1996).

Pengujian keabsahan pengguna ketiga perangkat lunak yang dikembangkan dalam penelitian ini menggunakan nama dan *password*. Karena perangkat lunak untuk administrator jaringan dan petugas administrasi akademik berisi menu yang dapat digunakan untuk memodifikasi data administrasi akademik, baik secara langsung maupun tidak langsung, maka pengujian keabsahan penggunaanya ditambah dengan alamat IP komputer. Alamat IP ini bersifat unik, yaitu di Internet tidak mungkin ada alamat IP yang sama (Tanenbaum dan Priatna, 1997). Dengan demikian maka kedua perangkat lunak tersebut tidak dapat diakses dari sembarang komputer sehingga data lebih aman.

Kesimpulan

Sistem yang diharapkan ternyata dapat dibangun. Perangkat lunak yang ada dapat mengetahui keabsahan penggunaanya. Perangkat lunak untuk administrator jaringan hanya dapat diakses dari *server* dan dapat digunakan untuk memberi kewenangan kepada petugas administrasi akademik. Perangkat lunak untuk petugas administrasi akademik hanya dapat diakses dari komputer petugas serta dapat digunakan untuk memberi kewenangan kepada mahasiswa dan memodifikasi data administrasi akademik. Perangkat lunak untuk mahasiswa dapat digunakan untuk mengisi KRS,

melihat KHS, dan melihat transkrip, atau melihat KHS dan melihat transkrip. Menu tersebut dapat berubah sendiri berdasarkan waktu di komputer *server* dan pernah tidaknya mahasiswa mengisi KRS. Semester KRS yang akan diisi dan kurikulum yang berlaku bagi mahasiswa dapat dideteksi secara otomatis. Pengisian KRS hanya dapat dilakukan sekali setiap semester dengan jumlah SKS maksimum sesuai aturan yang berlaku.

Daftar Pustaka

- Alter, S. (1992). *Information systems a management perspective*. Addison-Wesley Publishing Co.
- Amundsen, M. dan Smith, C. (1996). *Teach yourself database programming with visual basic 4 in 21 days*. Indianapolis: Sams Publishing.
- Brian, B. C., Shaw, J., McDonell, J., Smith, P., Slack, T., Dyson, N., Smith, J. M., Harrison, I, dan Matthews, S. (1994). *Webster's concise encyclopedia*. ATTICA Cybernetics.
- ExperTelligence.Inc. (1997). *WebBase user's guide version 4.10 build 56*. Santa Barbara: ExperTelligence.Inc.
- Hughes Technologies. (1997). *Mini SQL, 2.0 user guide*. Queensland: Hughes Technologies.
- Ladd, E., O'Donnell, J., Ablan, J., Anthony, T., Doherty, D., Dwight, J., Ellsworth, M., Erwin, M., Greene, S., Jung J., Knauss, G., Lockwood, T., Morgan, M., Niles, R., Roehl, B., Maria, P. S., Sutter, R. (1996). *Platinum edition using HTML 3.2, Java 1.1, and CGI*. Indianapolis: Que Corporation.

*Pengembangan Sistem Pemrosesan Transaksi Administrasi Akademik
Perguruan Tinggi Melalui Internet*

Morgan, M., Wandling, J., dan Casselberry, R. (1996). *Webmaster expert solutions*. Indianapolis. Que Corporation.