

Rancang Bangun Aplikasi Migrasi Data Studi Kasus *Smart Electronic Office (SEO)* Jurusan Teknologi Informasi Universitas Udayana

Kadek Wahyu Mega Duta, I Nyoman Piarsa, I Made Sukarsa

Jurusan Teknologi Informasi, Fakultas Teknik, Universitas Udayana

e-mail : wahyumegaduta@gmail.com, manpits@gmail.com, sukarsa@ee.unud.ac.id

Abstrak

Jurusan Teknologi Informasi Fakultas Teknik Universitas Udayana mengembangkan sebuah sistem informasi baru yang diberi nama *Smart Electronic Office (SEO)*. Sistem informasi tersebut merupakan penggabungan dan pengembangan dari beberapa sistem informasi lama (salah satunya SIMAK). Pengembangan ini memiliki tujuan utama untuk menciptakan sebuah sistem informasi baru, dimana terdapat integrasi antar modul-modul yang terpisah. Namun, perancangan dan pembangunan SEO menimbulkan masalah baru dari sisi *Database*. *Database* schema yang dimiliki SEO memiliki struktur yang berbeda dengan *Database* schema SIMAK. Hal ini akan menimbulkan kesulitan untuk melakukan migrasi data dari sistem SIMAK ke SEO. Aplikasi migrasi data ini telah dilengkapi dengan fitur pemetaan yang mampu memetakan *Database* schemahingga ke level field. Aplikasi ini merupakan aplikasi berbasis web. Kecepatan migrasi dari aplikasi ini sangat bergantung dengan kecepatan koneksi *Internet*. Untuk proses migrasi data, aplikasi ini menggunakan konsep ETL, dimana data yang dimigrasi akan melalui tiga tahapan, yaitu ekstraksi data, transformasi data dan loading data.

Kata kunci : SEO, *Database schema*, *schema mapping*, migrasi data

Abstract

Department of Information Technology in Udayana University is on develops a new information system called Smart Electronic Office (SEO). This new information system is an innovation of some old information system (such as SIMAK). The main purpose is to create a new information system with a powerful integration between separated modules. However, the design and development of SEO raises new issues on Database side. SEO's Database schema has a different structure with SIMAK's Database schema. This will make it difficult to perform the data migration from SIMAK to SEO. Application data migration has been equipped with a mapping feature that is able to map a Database schema to the field level. This application is a web base application. Migration speed of the application is very dependent on the speed of the Internet connection. For the data migration process, this application uses the ETL concept, where the data will be migrated through three stages, data extraction, data transformation and data loading.

Keyword : SEO, *Database schema*, *schema mapping*, data migration

1. Pendahuluan

Kebutuhan pengembangan aplikasi umumnya terjadi karena adanya perubahan konsep, alur dan proses bisnis yang terjadi di lingkungan sistem itu sendiri. Pengembangan aplikasi menjadi suatu keharusan ketika aplikasi tersebut sudah tidak dapat menunjang kebutuhan sistem secara keseluruhan.

Jurusan Teknologi Informasi Fakultas Teknik Universitas Udayana mengembangkan sebuah sistem informasi baru yang mengedepankan integrasi sistem yang handal. Sistem informasi tersebut merupakan penggabungan dan pengembangan dari beberapa sistem informasi lama (salah satunya SIMAK) yang telah digunakan kurang lebih selama sembilan

tahun. Pengembangan ini memiliki tujuan utama untuk menciptakan sebuah sistem informasi baru, dimana terdapat integrasi antar modul-modul yang terpisah, sehingga tidak memerlukan aplikasi ketiga (*middleware*) untuk melakukan komunikasi antarmodul. Berdasarkan kecerdasan dan kehandalan yang dikedepankan oleh sistem tersebut, maka sistem informasi tersebut kemudian diberi nama *Smart Electronic Office* (SEO).

Perancangan dan pembangunan SEO menimbulkan masalah baru dari sisi pengembangan *Database*. *Database* schema yang dimiliki SEO memiliki struktur yang berbeda dengan *Database* schema SIMAK. Hal ini akan menimbulkan masalah dalam melakukan migrasi data dari sistem SIMAK ke sistem SEO. Perlu dilakukan *schema mapping* sebelum melakukan migrasi data ke sistem SEO. Sehingga, diperlukan sebuah aplikasi yang memiliki fitur utama *schema mapping* dan fitur migrasi data, agar pengembangan SIMAK ke SEO dapat dilakukan tanpa ada data historis yang terbuang.

2. Metodologi Penelitian

Pembuatan Aplikasi Migrasi Data ini terdiri dari beberapa tahapan, yaitu: pendefinisian masalah, pengumpulan studi literatur, pengumpulan data, pembuatan aplikasi, pengujian sistem, dan penyusunan laporan hasil penelitian.

2.1 Pendefinisian Masalah

Pendefinisian masalah yang akan ditangani sistem yaitu struktur *Database* SIMAK dan struktur *Database* sistem SEO

2.2 Pengumpulan Studi Literatur

Pengumpulan teori-teori didapatkan dari buku, jurnal ilmiah baik nasional maupun internasional, serta modul-modul program yang menunjang penelitian ini

2.3 Pengumpulan Data

Pengumpulan data berupa dua buah *Database* yang akan digunakan untuk uji coba migrasi, yaitu *Database* SIMAK dan *Database* SEO.

2.3 Pembuatan Aplikasi

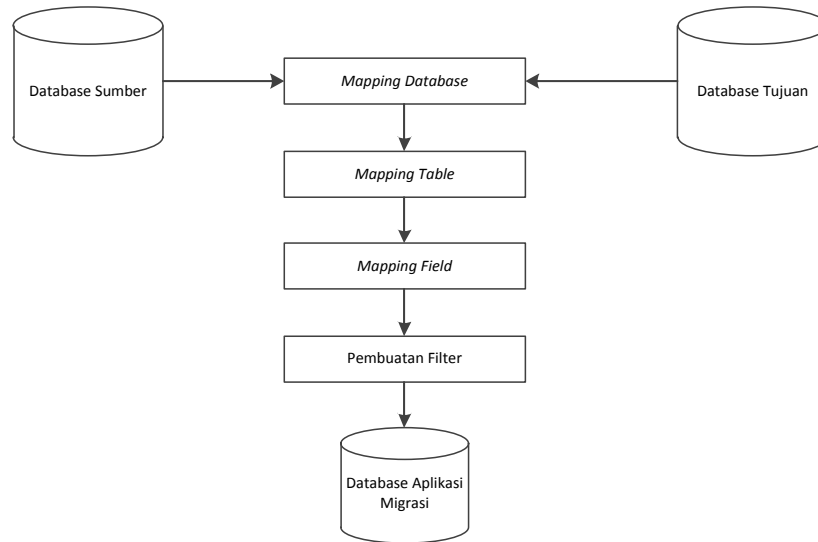
Pembuatan aplikasi meliputi pengembangan perangkat lunak memanfaatkan bahasa pemrograman *PHP* (*Hypertextpreprocessor*) dan *javascript*. Dilengkapi dengan perancangan *userinterface* menggunakan *HTML* (*Hyper Text Markup Language*) dan *CSS* (*Cascading Style Sheets*). *DBMS* (*Database Management System*) yang digunakan untuk melengkapi perancangan dan pembangunan sistem ini adalah *MySQL*.

2.3.1 Perancangan Sistem

Proses migrasi data terdiri dari dua sub sistem, yaitu pemetaan dan migrasi. Masing-masing sub sistem kemudian terbagi menjadi beberapa modul. Sub sistem pemetaan terdiri dari modul pemetaan *Database*, modul pemetaan tabel, modul pemetaan field, dan pembuatan *filter* data. Sedangkan, sub sistem migrasi terdiri dari modul ekstraksi data ke *database temporary* dan modul loading data ke *database* tujuan.

2.3.1.1 Sub-Sistem Pemetaan

Modul pemetaan terdiri dari empat tahapan utama, yaitu pemetaan *database*, pemetaan tabel, pemetaan *field*, dan pembuatan *filter* data.



Gambar 1. Gambaran Umum Sub-Sistem Pemetaan

Modul pertama dari sub-sistem pemetaan adalah pemetaan *database*, pada modul ini, dilakukan proses pembuatan profil koneksi. Profil koneksi yang dibuat adalah profil pemetaan terhadap dua buah *Database* yang akan menjadi *database* sumber data dan *database* tujuan migrasi.

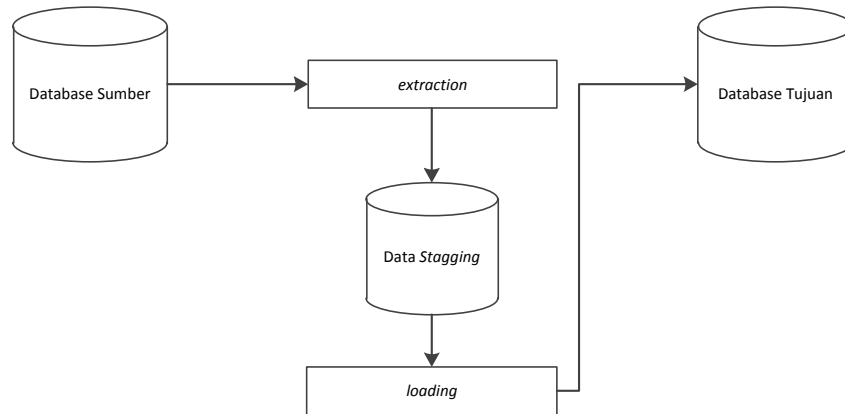
Modul kedua adalah pemetaan tabel, setelah *database* sumber dan *Database* tujuan dipilih, proses pemetaan tabel dilakukan dengan memilih tabel-tabel yang ada di sistem sumber dan memetakannya dengan tabel-tabel di sistem tujuan. Terdapat dua jenis pemetaan level tabel, yaitu: "pemetaan" dan "lookup".

Modul ketiga adalah pemetaan level *field* yang merupakan lanjutan dari proses pemetaan level tabel dengan jenis "pemetaan". Pemetaan level *field* bertujuan untuk memetakan *field* sumber dengan *field* tujuan. Pemetaan ini bersifat *onetoone*, untuk satu *field* sumber hanya dapat dipetakan ke satu *field* tujuan. Pemetaan *field* memiliki berbagai macam jenis konfigurasi. Tujuan konfigurasi ini adalah untuk mengurangi kesalahan pengiriman data, dengan menyamakan atribut-atribut yang terdapat pada kedua *field*. Sedangkan untuk pemetaan jenis "lookup" dilakukan dengan memetakan data yang ada di tabel sumber dengan data yang ada di tabel tujuan. Pemetaan data akan menghasilkan sebuah tabel baru yang berfungsi sebagai tabel *reference* untuk pemetaan *field*. Tabel baru yang dihasilkan berisi dua data dari tabel sumber dan tabel tujuan, dan data id baru, sebagai tanda pengenal baru.

Modul terakhir adalah pembuatan *filter* data, proses pembuatan *filter* data bertujuan untuk menyeleksi data sebelum dilakukannya tahapan migrasi. Proses *filter* data dilakukan pada pemetaan level tabel. Sehingga tiap-tiap pemetaan level tabel memiliki *filter* data yang berbeda.

2.3.1.2 Sub-Sistem Migrasi

Sub-sistem migrasi merupakan subsistem utama yang terdiri dari dua tahapan, yaitu tahapan ekstraksi data, dan tahapan *loading* data.



Gambar 2. Gambaran Umum Sub-Sistem Migrasi

Tahapan migrasi data diawali dengan proses ekstraksi data. Proses ekstraksi data adalah proses dimana data dari sistem sumber diekstrak ke tabel *temporary*. Proses ekstraksi pada aplikasi ini berbeda dengan proses ekstraksi pada aplikasi ETL. Dimana, pada aplikasi ini, data yang diekstrak telah melalui proses transformasi. Data melalui tahapan transformasi ketika data diambil dari sistem sumber dan ditransfer menuju tabel *temporary*. Jadi data yang telah diekstraksi adalah data yang telah melalui proses transformasi melalui *queryselect* dan proses *filter data*.

Data yang telah diekstrak menuju tabel *temporary* selanjutnya akan melalui proses *loading data*. Proses *loading data* merupakan tahapan akhir dari proses migrasi data, dimana data dimigrasi dari tabel *temporary* menuju sistem tujuan.

3. Kajian Pustaka

Pengumpulan teori-teori yang didapatkan dari buku atau *Internet* serta modul-modul program yang menunjang penelitian ini.

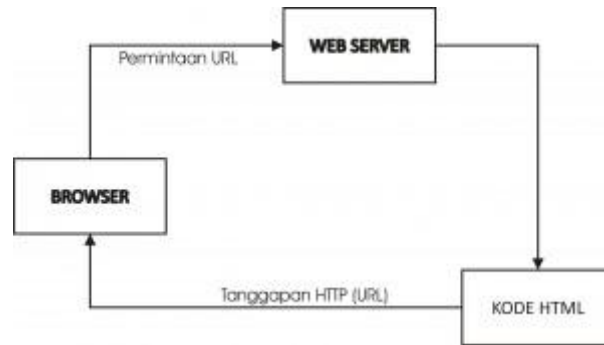
3.1 Pemrograman Berbasis Web

Konsep dari pemrograman berbasis web merupakan konsep kerjasama antara permintaan (*request*) dan umpan balik (*feedback*) dari *client* atau *user* ke *web service* atau *server* melalui *web browser*. Kerjasama dalam hal ini adalah konsep *client server* dan *peer to peer*.

Konsep *client server* adalah koneksi dan komunikasi yang dilakukan dua komputer dimana satu sisi bertindak sebagai *client* dan sisi lain sebagai *server*. Tugas dari server hanya sebagai alat penanggap permintaan *client*. *Client* bertugas sebagai pengirim suatu permintaan suatu proses kepada *server*.

3.1.1 Client Side Scripting

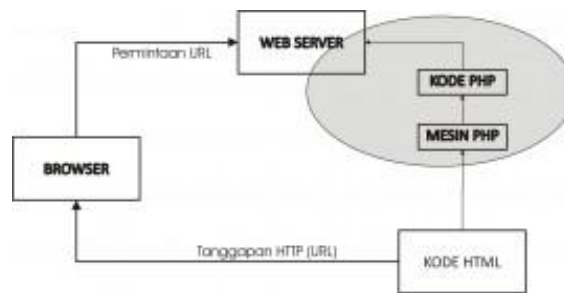
Cara kerja *Client Side Scripting* berawal dari *client* yang memanggil berdasarkan URL (*Uniform Resource Locator*) melalui *browser*, kemudian *browser* mendapat alamat dari *webserver*, yang nantinya akan memberikan segala informasi yang dibutuhkan *webbrowser*. *Webbrowser* yang sudah mendapat informasi segera melakukan proses penerjemahan kode HTML (*Client Side Scripting*) dan menampilkannya ke layar pemakai.



Gambar 3. Ilustrasi client side scripting

3.1.2 Server Side Scripting

Server Side Scripting merupakan sebuah teknologi *scripting* atau pemrograman web dimana *script* (program) dikompilasi atau diterjemahkan di server. Dengan *serversidescripting*, memungkinkan untuk menghasilkan halaman web yang dinamis [1].



Gambar 4. Ilustrasi server side scripting

3.2 Database

Database adalah dua atau lebih simpanan data dengan elemen-elemen data penghubung, yang dapat diakses lebih dari satu cara. *Database* dinyatakan dengan teknik-teknik formal dan manajemen *database*. Dari definisi diatas, maka dapat dikatakan bahwa basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya [2].

Untuk mengelola *database* diperlukan suatu perangkat lunak yang disebut DBMS (*Database Management System*). DBMS merupakan suatu sistem perangkat lunak yang memungkinkan *user* (pengguna) untuk membuat, memelihara, mengontrol, dan mengakses *Database* secara praktis dan efisien. Dengan *DBMS*, *user* akan lebih mudah mengontrol dan memanipulasi data yang ada [3].

3.3 Migrasi Data

Migrasi adalah pergerakan suatu "file system" dari satu server ke server yang lain. Migrasi *Database* menunjuk pada koleksi proses-proses dan prosedur-prosedur untuk mengkonversi data dari satu *server database* ke server *database* yang lainnya [4].

3.4 ETL

Dalam *Data Warehouse* ETL (*extraction, transformation, loading*) merupakan aplikasi yang terpisah dari *data warehouse* dan berfungsi sebagai pondasi dari *data warehouse* itu sendiri. ETL terdiri dari tiga proses utama. Ketiga proses ini dilakukan secara berurutan. Keberhasilan suatu proses dalam ETL ini tergantung dari proses ETL sebelumnya [5].

3.4.1 Extraction

Data mentah yang berasal dari sistem informasi operasional/sistem sumber biasanya ditulis atau di-copy langsung ke dalam media penyimpanan/*staging area* dengan restrukturisasi

seminimal mungkin. Hal ini dilakukan untuk menjaga keaslian data yang didapat dari sistem sumber. Data yang telah diambil dalam proses ekstraksi ini dapat dibaca beberapa kali sesuai dengan keperluannya. Dalam beberapa kasus, data hasil ekstraksi ini bisa dihapus setelah dilakukannya proses transformation karena dianggap sudah tidak berguna lagi. Kemudian dalam kasus lain, data ekstraksi ini bisa disimpan sebagai arsip cadangan jangka panjang namun memerlukan *space* yang besar [5].

3.4.2 Transformation

Perubahan sekecil apapun yang dilakukan pada data mentah hasil ekstraksi adalah transformasi. Misalnya melakukan proses seleksi dari data yang mengandung nilai *null*. Jika data ditemukan *null* maka data akan dihapus. Contoh diatas merupakan contoh kecil dari sebuah proses transformasi yang dilakukan oleh ETL. Beberapa hal penting yang sering dilakukan dalam tranformasi ini dan menjamin data yang akan diolah sudah bersih dari data yang dianggap sampah atau tidak perlu adalah *cleaning* dan *conforming* [5].

1. *Cleaning*: dalam kebanyakan kasus, tingkat kualitas data pada sistem-sistem sumber berbeda-beda. Kualitas data sistem sumber ini juga berbeda dengan kualitas data yang dibutuhkan pada *data warehouse* itu sendiri. Bertolak dari hal tersebut maka pengolahan data dapat melibatkan banyak proses-proses terpisah antara lain memeriksa nilai-nilai yang valid, memastikan konsistensi dari nilai-nilai tersebut, membuang duplikasi atau redudansi dari data.
2. *Conforming*: data yang telah bersih kemudian akan dicek lagi sebelum dilakukan proses berikutnya. Proses ini adalah memisahkan data sumber yang identik atau jika menggunakan hitungan *numeric* data yang tidak termasuk dalam *range* tertentu. *Conforming* ini membutuhkan suatu kesepakatan dari pengguna data untuk menentukan data mana saja yang akan digunakan dalam *data warehouse*.

3.4.3 Loading

Proses *loading* atau dikenal juga dengan proses *delivering* adalah suatu proses dimana data hasil transformasi siap untuk dimasukkan ke dalam *datawarehouse* itu sendiri. Pembentukan struktur tabel dari data yang akan di-loading merupakan tugas dari *designer* dari *datawarehouse* itu sendiri. *Loading* ini merupakan langkah akhir dan penting dalam menjamin ketersediaan data dalam *datawarehouse*.

3.5 SIMAK

Sistem Informasi Akademik (SIMAK) merupakan wadah bagi civitas akademika di lingkungan Universitas Udayana, terutama mahasiswa dan dosen untuk mengakases berbagai data akademik. Pengaksesan data berupa meng-entry biodata dan mengisi dan atau memperbaiki KRS serta melihat berbagai data akademik seperti KHS, Transkrip, jadwal pengisian/perbaikan KRS dan matakuliah, dan berbagai informasi/pemberitahuan akademik.

3.6 SEO

Smart E-Office (SEO) merupakan sebuah sistem terintegrasi yang dikembangkan oleh Teknologi Informasi Universitas Udayana sebagai media penerapan teknologi dalam menjalankan berbagai fungsi-fungsi yang ada dengan mengedepankan tampilan yang menarik, desain *layout*, responsibilitas terhadap berbagai *device*, kehandalan sistem, kelengkapan fitur dan pengaplikasian yang mudah. SEO Teknologi Informasi Unud terdiri atas 9 Sistem Informasi Managemen (SIM) atau modul yang saling terintegrasi satu sama lain yaitu akademik, kepegawaian, kemahasiswaan, penjadwalan, absensi, penelitian, *tracer study*, bimbingan konseling, dan layanan SMS. Semua modul tersebut dihubungkan dengan sebuah *Single Sign On* (SSO) sebagai sentral data pengguna untuk dapat berpindah dari satu modul ke modul yang lainnya hanya dengan sekali melakukan login. Integrasi antar modul akan selalu terjadi, mengingat beberapa modul memerlukan data yang bersumber dari modul lainnya. Sehingga tidak dapat berjalan sendiri. Integrasi data dengan semua modul yang ada, dibatasi sesuai dengan aturan-aturan terkait pengguna yang melakukan login serta hak akses yang diperuntukkan.

SEO Teknologi Informasi Unud memberikan kemudahan dalam melakukan berbagai kegiatan terkait modul yang dipilih sehingga mengurangi aktifitas dan fungsi manual yang ada

serta mengurangi kesalahan yang disebabkan human *error*. Penundaan terhadap suatu tugas dapat meminimalisir karena telah dikerjakan secara otomatis oleh sistem. Hal ini akan meningkatkan produktifitas Teknologi Informasi dalam melakukan berbagai aktifitas akademik, kepegawaian, kemahasiswaan, penelitian sampai kepada bimbingan dan konseling tanpa harus melalui tatap muka dan menghabiskan kertas fisik untuk melakukan audiensi ataupun asistensi.

Umumnya, sistem yang ada di lingkungan Universitas Udayana terdiri dari modul-modul yang terpisah dan berjalan sendiri. Sehingga tidak ada integrasi data dan biasanya memerlukan aplikasi ketiga sebagai *middleware*. Teknologi Informasi menghadirkan SEO sebagai suatu terobosan baru dalam hal integrasi sistem yang kompleks untuk dapat menjawab kebutuhan akan sistem yang dapat bekerja dan membantu dalam menjalankan berbagai fungsi yang ada dalam Teknologi Informasi dengan baik dan maksimal.

4. Hasil dan Pembahasan

Untuk mengetahui keberhasilan proses migrasi data, perlu dilakukan uji coba migrasi. Data yang diuji coba adalah data penawaran mata kuliah dan data KRS 2008. Migrasi dilakukan dari *Database SIMAK ke Database SIMAK-SEO*.

Berikut ini merupakan data penawaran matakuliah tahun 2008 untuk jurusan Teknologi Informasi. Terdapat 98 baris data yang diambil dari tabel *tb_matakuliah_ditawarkanDatabase db_simak_ft*.

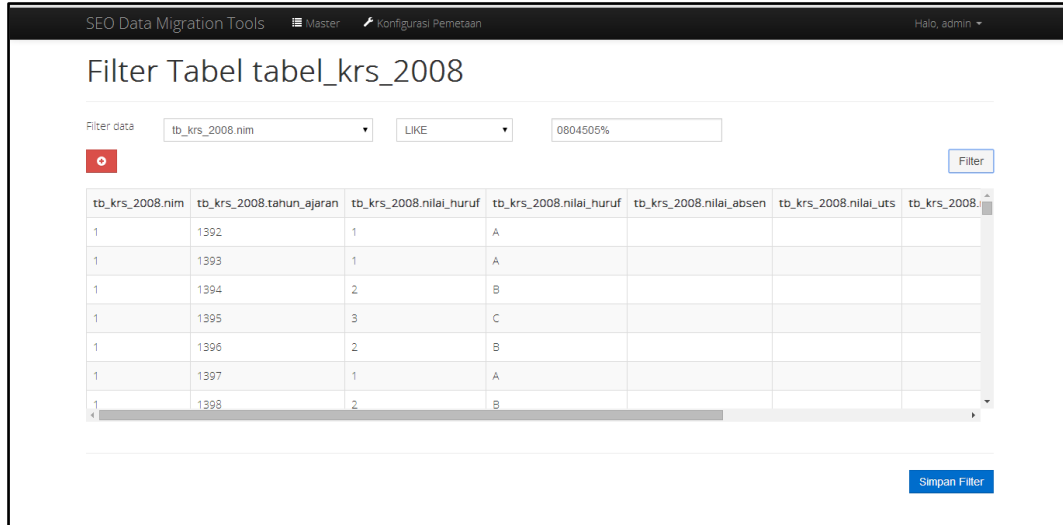
tahun_ajaran	id_matakuliah_ditawarkan	id_jurusan	semester	kode_matakuliah	kelas	jumlah_rencana_tatap_muka	jumlah_realisasi_tata
2008	670	5	1	TI1101	A	0	0
2008	668	5	1	TI1102	A	0	0
2008	672	5	1	TI1103	A	0	0
2008	680	5	1	TI1104	A	0	0
2008	663	5	1	TI1105	A	0	0
2008	674	5	1	TI1106	A	0	0
2008	678	5	1	TI1107	A	0	0
2008	676	5	1	TI1108	A	0	0
2008	667	5	1	TI1109	A	0	0
2008	669	5	1	TI1102	B	0	0
2008	671	5	1	TI1101	B	0	0
2008	673	5	1	TI1103	B	0	0
2008	675	5	1	TI1106	B	0	0
2008	677	5	1	TI1108	B	0	0
2008	679	5	1	TI1107	B	0	0
2008	681	5	1	TI1104	B	0	0
2008	1125	5	2	TI1101		0	0
2008	1126	5	2	TI1102		0	0
2008	1127	5	2	TI1103		0	0
2008	1128	5	2	TI1104		0	0

Gambar 5. Data tabel *tb_matakuliah_ditawarkan*

Proses migrasi data tabel *tb_matakuliah_ditawarkan* dilakukan terlebih dahulu, karena untuk seleksi tabel *tb_krs_2008* memerlukan data yang telah dimigrasi ke tabel *tb_mk_penawaran (SIMAK-SEO)*. Berikut merupakan proses migrasi data tabel *tb_matakuliah_ditawarkan*.

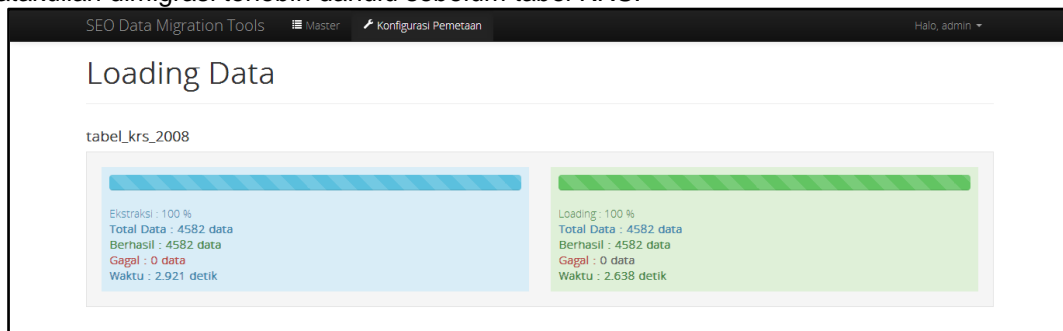
Gambar 6. Proses migrasi data tabel *tb_matakuliah_ditawarkan*

Tabel pada `tb_krs_2008` berisikan data KRS mahasiswa 2008 di seluruh jurusan yang ada di Fakultas Teknik. Untuk menyeleksi data KRS mahasiswa TI angkatan 2008, diperlukan sebuah *filter*. Gambar 7 menunjukkan proses *filter* data untuk tabel `tb_krs_2008`.



Gambar 7. *Filter* tabel `tb_krs_2008`

Gambar 8 menunjukkan proses migrasi data pada tabel `tb_krs_2008`. Proses migrasi untuk tabel KRS angkatan lain berlangsung sama seperti tabel `tb_krs_2008`. Tabel penawaran matakuliah dimigrasi terlebih dahulu sebelum tabel KRS.



Gambar 8. Proses migrasi data tabel `tb_krs_2008`

Tabel 1 menunjukkan keberhasilan proses migrasi data KRS mahasiswa dari tahun 2008 hingga tahun 2014.

Tabel 1. Data keberhasilan proses migrasi

Tabel	Total Data	Berhasil	Gagal
<code>tb_krs_2008</code>	4582 data	4582 data	0 data
<code>tb_krs_2009</code>	4532 data	4532 data	0 data
<code>tb_krs_2010</code>	4247 data	4247 data	0 data
<code>tb_krs_2011</code>	5914 data	5914 data	0 data
<code>tb_krs_2012</code>	3661 data	3661 data	0 data
<code>tb_krs_2013</code>	2977 data	2977 data	0 data
<code>tb_krs_2014</code>	936 data	936 data	0 data

5. Kesimpulan

Berdasarkan rumusan masalah serta uraian pembahasan dan analisis hasil dapat disimpulkan beberapa hal sebagai berikut :Migrasi data antar *Database* yang memiliki struktur berbeda memerlukan tahapan pemetaan untuk struktur *Database*. Pemetaan dilakukan mulai

dari level *Database*, level tabel, hingga ke level paling detail, yaitu level *field*. Pemetaan level *field* memiliki banyak kemungkinan berdasarkan kecocokan *attribute*, sehingga memerlukan fitur konfigurasi untuk mengantisipasi kegagalan migrasi. Penerapan konsep ETL dalam proses migrasi data sangat membantu dalam memperbaiki kualitas data. Data yang berada pada sistem sumber, dapat disaring melalui proses ETL, sehingga proses migrasi data yang dilakukan dari sistem SIMAK ke sistem SEO, menghasilkan data yang memiliki kualitas lebih baik. Keberadaan data *staging* juga sangat membantu dalam menjaga konsistensi data sumber.

Daftar Pustaka

- [1] Solichin, A. Pemrograman Web dengan PHP dan MySQL. Jakarta: Universitas Budi Luhur. 2010.
- [2] Iskandar, A., Rangkuti, H. Perancangan Sistem Informasi Penjualan Tunai pada PT. Klaten Bercahaya. Jurnal Basis Data, ICT Research Center UNAS. Vol.3 No.2 Nopember 2008. ISSN 1978-9483. 126.
- [3] Solichin, A. MySQL 5 dari Pemula Hingga Akhir. Jakarta: Universitas Budi Luhur. 2010.
- [4] Ricky Y. Aplikasi Migrasi *Database* Dan Replikasi Bi-Directional. Konferensi Nasional Sistem dan Informatika. 2011.KNS&I11-038. 246.
- [5] Kimball, R., Caserta, J. The Data Warehouse ETL Toolkit Practical Techniques for Extraction, Cleaning, Conforming and Delivering Data. Canada: Wiley Publishing. Inc. 2004.