


PEN

 PERPUSTAKAAN	UNITAS PERPUSTAKAAN UNIVERSITAS JENDERAL SOEDHARTO YOGYAKARTA
Tanggal	- 8 MAR 2011
Fingerprint	26/TIF/Hel.3/Pen/2011
Klasifikasi	: 005.12 / Pur / p
Subjek	: software system and Design



# LAPORAN AKHIR

## PENELITIAN

PEMBANGUNAN PERANGKAT LUNAK ALAT BANTU PROSES  
INTEGRASI SISTEM INFORMASI AKADEMIK DAN SITUS KULIAH  
(STUDI KASUS: UNIVERSITAS ATMA JAYA YOGYAKARTA)

PENELITIAN LABORATORIUM/LAPANGAN



Oleh:

**Yohanes Sigit Purnomo W.P., S.T., M.Kom.**

**Fakultas Teknologi Industri  
Universitas Atma Jaya Yogyakarta**

**2011**

## LEMBAR PENGESAHAN

1.	a. Judul Penelitian	:	PEMBANGUNAN PERANGKAT LUNAK ALAT BANTU PROSES INTEGRASI SISTEM INFORMASI AKADEMIK DAN SITUS KULIAH (STUDI KASUS: UNIVERSITAS ATMA JAYA YOGYAKARTA)
	b. Macam Penelitian	:	Laboratorium
2.	Peneliti		
	a. Nama	:	Y. Sigit Purnomo W.P., S.T., M.Kom.
	b. Jenis Kelamin	:	Laki-laki
	c. Usia saat pengajuan proposal	:	31 tahun 11 bulan
	d. Jabatan Akademik/Gol	:	Lektor / IIIc
	e. Fakultas / Program Studi	:	Teknologi Industri / Teknik Informatika
3.	Jumlah Peneliti	:	1 (satu) orang
4.	Lokasi Penelitian	:	Yogyakarta
5.	Jangka Waktu Penelitian	:	6 (enam) bulan
6.	Biaya yang disetujui	:	2.850.000,- (dua juta delapan ratus lima puluh ribu rupiah)

Yogyakarta, 28 Februari 2011

Ketua Peneliti,

Y. Sigit Purnomo W.P., ST, M.Kom.

Mengetahui,  
Kepala Laboratorium Jaringan Komputer

Th. Dewi Indriasari, S.T., M.Sc.

Dehan FTI UAJY,  
  
Dr. P. Kristyanto M.Eng., Ph.D.

Dr. MF. Shellyana Junaedi, S.E.,  
  
Dr. MF. Shellyana Junaedi, S.E.

## DAFTAR ISI

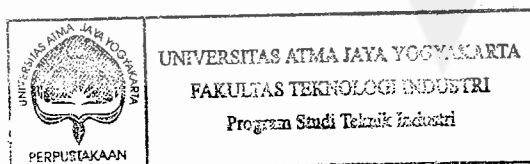
LEMBAR PENGESAHAN .....	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	v
DAFTAR TABEL.....	vi
INTISARI .....	vii
1. PENDAHULUAN .....	1
1.1 Pendahuluan.....	1
1.2 Perumusan Masalah.....	2
1.3 Tujuan Penelitian .....	2
1.4 Manfaat Hasil Penelitian.....	2
1.5 Metodologi.....	2
2. LANDASAN TEORI.....	4
2.1 Definisi Integrasi Aplikasi.....	4
2.2 Tingkatan Integrasi Aplikasi .....	4
2.3 Three-tier Client-server Architecture .....	7
3. ANALISIS DAN PERANCANGAN KEBUTUHAN INTEGRASI .....	11
3.1 Analisis Kebutuhan Integrasi .....	11
3.1.1 Spesifikasi Struktur Data SIATMA.....	12
3.1.2 Spesifikasi Struktur Data Situs Kuliah.....	13
3.1.3 Mapping Struktur Data SIATMA dan Situs Kuliah.....	14
3.2 Perancangan Kebutuhan Integrasi.....	17
4. ANALISIS DAN PERANCANGAN SISTEM .....	18
4.1 Spesifikasi Kebutuhan Fungsional .....	18
4.2 Spesifikasi Kebutuhan Data.....	18
4.3 Spesifikasi Kebutuhan Antarmuka Eksternal .....	20
4.4 Perancangan Fungsional.....	21
4.5 Perancangan Data .....	21
4.6 Perancangan Arsitektur Navigasi Antar Muka Pengguna .....	23
5. PROSES INTEGRASI SISTEM .....	24
5.1 Proses Generate File.....	24
5.2 Proses Bulk Course Upload .....	26

6. KESIMPULAN.....27  
DAFTAR PUSTAKA.....28



## DAFTAR GAMBAR

Gambar 2.1. Contoh Proses Bisnis Sederhana yang Membutuhkan Integrasi Aplikasi (Sumber: Microsoft Corporation).....	5
Gambar 2.2. Kapabilitas Integrasi pada Tingkatan Data (Sumber: Microsoft Corporation).....	6
Gambar 2.3. Intergrasi pada Tingkat Data (Sumber: Juric et all., 2007) .....	6
Gambar 2.4. Arsitektur two-tier .....	8
Gambar 2.5. Three-tier client-server architecture.....	9
Gambar 2.6. Konsep three-tier [SAD02]. .....	9
Gambar 3.1. Spesifikasi Struktur Data SIATMA .....	12
Gambar 3.2. Spesifikasi Struktur Data SITUS KULIAH.....	13
Gambar 3.3. Rancangan Arsitektur Integrasi Aplikasi .....	17
Gambar 4.1. Diagram Use Case SIATMA2KULIAH .....	18
Gambar 4.2. Diagram Kelas SIATMA2KULIAH.....	21
Gambar 4.3. Rancangan Antar Muka Pengguna SIATMA2KULIAH: Form Utama .....	23
Gambar 5.1. DataSet SIATMA .....	24
Gambar 4.2. DataSet Struktur File.....	25
Gambar 4.3. DataSet Struktur File.....	26



## DAFTAR TABEL

<b>Tabel 3.1. Mapping Struktur Data SIATMA dan Situs Kuliah .....</b>	<b>14</b>
<b>Tabel 3.2. Nilai Default pada Struktur Data Situs Kuliah .....</b>	<b>15</b>
<b>Tabel 4.1. Use Case Glossary SIATMA2KULIAH.....</b>	<b>18</b>
<b>Tabel 4.2. Tabel Spesifikasi Kebutuhan Data.....</b>	<b>19</b>
<b>Tabel 4.3. Dekomposisi Data File.....</b>	<b>21</b>



## INTISARI

Universitas Atma Jaya Yogyakarta memiliki Sistem Informasi Akademik (SIATMA) yang digunakan untuk mendukung proses administrasi kegiatan akademik seperti penawaran kelas, pendaftaran kelas, pengaturan ujian tengah dan akhir semester, pemasukan data nilai, dan kegiatan akademik lainnya . Universitas Atma Jaya juga telah mengimplementasikan Situs Kuliah Universitas Atma Jaya Yogyakarta yang sudah diintegrasikan dengan layanan Live@edu hasil kerjasama dengan PT. Microsoft Indonesia. Kedua sistem tersebut memiliki basisdata sendiri-sendiri yang platform dan struktur basisdatanya berbeda.

SIATMA dan Situs Kuliah meski platform dan struktur basisdatanya berbeda, sebenarnya ada beberapa data yang ada di kedua sistem tersebut yang saling terkait, seperti data mata kuliah, data pengajar, dan data peserta kuliah (mahasiswa/i). Hal ini menunjukkan bahwa dapat dilakukan integrasi pada tingkatan data antara kedua sistem tersebut. Cara yang dapat dilakukan untuk melakukan integrasi pada tingkatan data adalah dengan menambahkan logic yang memungkinkan setiap aplikasi untuk memahami data yang datang dari aplikasi lainnya atau menambahkan logic yang memungkinkan setiap aplikasi menginterpretasikan data yang dihasilkan aplikasi ke dalam format tertentu dan diinterpretasiokan ke dalam format yang dipahami aplikasi.

Kata kunci: integrasi, sistem, data, SIATMA, Situs Kuliah



# BAB 1. PENDAHULUAN

## 1.1 Pendahuluan

Universitas Atma Jaya Yogyakarta saat ini telah mengimplementasikan Sistem Informasi Akademik (SIATMA) yang digunakan untuk mendukung proses administrasi kegiatan akademik seperti penawaran kelas, pendaftaran kelas, pengaturan ujian tengah dan akhir semester, pemasukan data nilai, dan kegiatan akademik lainnya. Dari 6 fakultas yang ada di Universitas Atma Jaya Yogyakarta, semua fakultas sudah menggunakan SIATMA.

Selain mengimplementasikan SIATMA, Universitas Atma Jaya Yogyakarta (Situs Kuliah) pada tahun ini juga telah mengimplementasikan Situs Kuliah Universitas Atma Jaya Yogyakarta yang sudah diintegrasikan dengan layanan Live@edu hasil kerjasama dengan PT. Microsoft Indonesia. Situs kuliah ini, nantinya diharapkan dapat digunakan oleh semua program studi yang ada di Universitas Atma Jaya Yogyakarta untuk mendukung proses pembelajaran. Sampai saat ini, sudah ada beberapa program studi yang mulai menggunakan situs kuliah ini.

Saat ini, kedua sistem tersebut (SIATMA dan Situs Kuliah), masing-masing memiliki basisdata sendiri-sendiri yang platform dan struktur basisdatanya berbeda. SIATMA menggunakan Microsoft SQL Server sebagai platform database management system (DBMS)-nya. Sedangkan Situs Kuliah menggunakan MySQL sebagai platform DBMS-nya. Meski platform dan struktur basisdatanya berbeda, sebenarnya ada beberapa data yang ada di kedua sistem tersebut yang saling terkait, seperti data mata kuliah, data pengajar, dan data peserta kuliah (mahasiswa/i).

Proses yang terjadi saat ini, pengelola Situs Kuliah memasukkan data-data mata kuliah dan data-data pengajar ke dalam Situs Kuliah secara manual menggunakan antarmuka sistem yang disediakan di Situs Kuliah. Data-data peserta mata kuliah tidak dimasukkan ke Situs Kuliah oleh pengelola Situs Kuliah melainkan mahasiswa diminta untuk melakukan proses enrolment sendiri-sendiri ke mata kuliah yang diambil di Situs Kuliah. Hal ini tentu saja dapat menimbulkan permasalahan bagi pengelola Situs Kuliah maupun Mahasiswa.

## 1.2 Perumusan Masalah

Rumusan masalah yang akan dijawab melalui penelitian yang akan dilakukan adalah sebagai berikut:

1. Bagaimana melakukan analisis kebutuhan dan perancangan integrasi antara SIATMA dan Situs Kuliah?
2. Bagaimana mengembangkan perangkat lunak yang dapat digunakan sebagai alat bantu untuk proses integrasi data SIATMA dan Situs Kuliah?

## 1.3 Tujuan Penelitian

Penelitian yang akan dilakukan bertujuan untuk:

1. Menganalisis kebutuhan integrasi dan merancang bagaimana proses integrasi data SIATMA dan Situs Kuliah.
2. Membangun suatu perangkat lunak yang dapat digunakan sebagai alat bantu untuk proses integrasi data SIATMA dan Situs Kuliah.

## 1.4 Manfaat Hasil Penelitian

Manfaat yang didapat dari penelitian yang akan dilakukan adalah sebagai berikut:

1. Pemahaman tentang proses analisis kebutuhan dan perancangan integrasi data SIATMA dan Situs Kuliah
2. Pemahaman tentang pembangunan perangkat lunak yang dapat digunakan sebagai alat bantu untuk proses integrasi data SIATMA dan Situs Kuliah.

## 1.5 Metodologi

Penelitian ini dilakukan dengan melakukan sejumlah aktivitas yang berkaitan, antara lain:

### 1. Studi Pustaka

Studi pustaka dilakukan dengan mengumpulkan informasi dari buku-buku, artikel, maupun jurnal ilmiah yang membahas mengenai hal-hal yang terkait dengan proses geocoding ataupun sistem berbasis layanan lokasi secara umum.

## 2. Analisis Kebutuhan Integrasi Sistem

Analisis kebutuhan integrasi sistem dilakukan dengan menggali karakteristik masing-masing sistem (SIATMA dan Situs Kuliah), khususnya dari sisi struktur data yang ada. Hal tersebut akan digunakan untuk menentukan teknik/metode integrasi apa yang tepat untuk diterapkan dalam kedua sistem tersebut.

## 3. Analisis Kebutuhan Sistem.

Analisis kebutuhan sistem dilakukan dengan menggali kebutuhan fungsional dari layanan aplikasi yang akan dikembangkan dan menentukan sejauh mana kebutuhan-kebutuhan tersebut akan diakomodasi dalam layanan aplikasi yang akan dibangun.

## 4. Perancangan Sistem.

Perancangan sistem dilakukan untuk mendapatkan deskripsi mengenai arsitektural layanan aplikasi dan deskripsi data.

## 5. Implementasi Sistem.

Implementasi sistem dilakukan dengan menterjemahkan deskripsi perancangan yang telah dibuat ke dalam kode-kode program sesuai dengan tools yang digunakan untuk membangun layanan aplikasi.

## 6. Pengujian Sistem

Pengujian sistem dilakukan untuk menguji fungsionalitas layanan aplikasi yang akan dibangun apakah sudah sesuai dengan spesifikasi yang telah ditentukan.

## 7. Penulisan Laporan dan Dokumentasi

Tahap ini dilakukan dengan membuat dokumentasi terhadap seluruh aktivitas penelitian dengan harapan dapat dipergunakan untuk penelitian lainnya.

## BAB 2. LANDASAN TEORI

### 2.1 Definisi Integrasi Aplikasi

Sebuah organisasi umumnya memiliki berbagai sistem informasi yang digunakan untuk mendukung proses bisnis yang terdapat di organisasi tersebut. Seringkali, pengembangan sistem informasi yang dilakukan tidak melalui perencanaan yang baik sehingga di organisasi tersebut terdapat pulau-pulau sistem informasi maupun data. Hal ini tentu saja akan menimbulkan berbagai permasalahan, seperti inkonsistensi data dan meningkatnya biaya pemeliharaan sistem. Salah satu cara yang dapat digunakan untuk mengatasi permasalahan tersebut adalah dengan melakukan integrasi aplikasi.

Integrasi aplikasi didefinisikan sebagai mekanisme berbagi pakai proses dan atau data dari berbagai aplikasi yang dimiliki oleh perusahaan secara aman dan teratur (Microsoft Corporation, 2003). Jika integrasi aplikasi berhasil dilakukan secara efektif, maka perusahaan atau organisasi akan memperoleh berbagai manfaat seperti aplikasi dapat dikenalkan ke organisasi secara efisien dan dengan biaya yang rendah, memungkinkan terjadinya modifikasi proses bisnis sesuai dengan kebutuhan organisasi, tersedianya berbagai *delivery channel* bagi organisasi, dan memungkinkan untuk menambahkan otomatisasi dalam proses bisnis yang sebelumnya masih membutuhkan intervensi manusia (Microsoft Corporation, 2003).

### 2.2 Tingkatan Integrasi Aplikasi

Integrasi aplikasi dapat dibedakan menjadi beberapa tingkatan, yaitu tingkatan penyajian/presentasi, tingkatan proses bisnis, tingkatan data, dan tingkatan komunikasi (Microsoft Corporation, 2003; Juric, et.all, 2007).

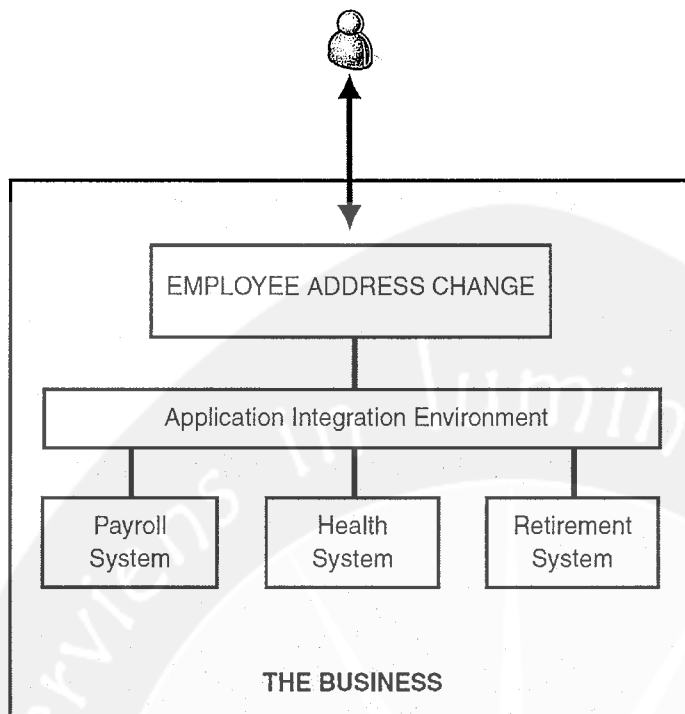
#### a. Tingkatan Penyajian/Presentasi

Integrasi pada tingkatan ini dapat menjadi cara yang sangat efektif untuk mengintegrasikan aplikasi di lingkungan organisasi. Hal ini karena memungkinkan berbagai macam aplikasi untuk ditampilkan sebagai satu aplikasi kohesif yang seringkali menggunakan antarmuka aplikasi yang sudah ada.

#### b. Tingkatan Proses Bisnis

Integrasi pada tingkatan proses bisnis seringkali menjadi titik awal untuk menentukan lingkungan integrasi aplikasi. Langkah ini diawali dengan

mendefinisikan proses bisnis kemudian menentukan integrasi logikan yang berhubungan dengan proses tersebut.



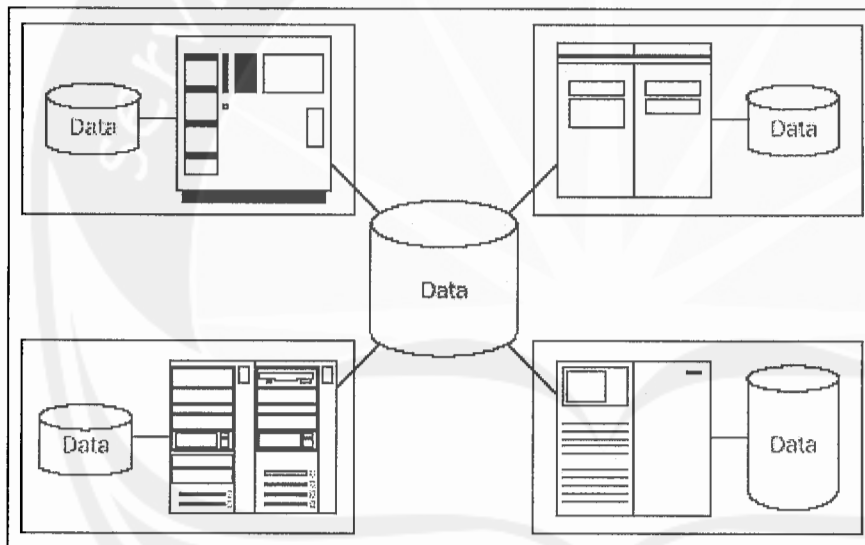
**Gambar 2.1. Contoh Proses Bisnis Sederhana yang Membutuhkan Integrasi Aplikasi**  
(Sumber: Microsoft Corporation)

c. Tingkatan Data

Integrasi pada tingkatan data diperlukan karena seringkali aplikasi yang berbeda juga memiliki cara yang berbeda ketika menangani data. Cara yang dapat dilakukan untuk melakukan integrasi pada tingkatan data adalah dengan menambahkan logic yang memungkinkan setiap aplikasi untuk memahami data yang datang dari aplikasi lainnya atau menambahkan logic yang memungkinkan setiap aplikasi menginterpretasikan data yang dihasilkan aplikasi ke dalam format tertentu dan diinterpretasikan ke dalam format yang dipahami aplikasi. Kapabilitas integrasi pada tingkatan data dapat dilihat pada Gambar 2.

Capability	Description
Data Transformation	Modifies the appearance or format of the data (without altering its content) so that applications can use it.
Data Validation	Determines whether data meets the predefined technical criteria.
Data Access	Determines how data is collected and presented to applications.
Schema Definition	Maintains predefined schemas.
Mapping	Manages the relationship between applications and determines how the data must be transformed from the source to the target application.
Schema Recognition	Checks that the schema can be read and understood, and that it is well-formed.

**Gambar 2.2. Kapabilitas Integrasi pada Tingkatan Data (Sumber: Microsoft Corporation)**



**Gambar 2.3. Intergrasi pada Tingkat Data (Sumber: Juric et al., 2007)**

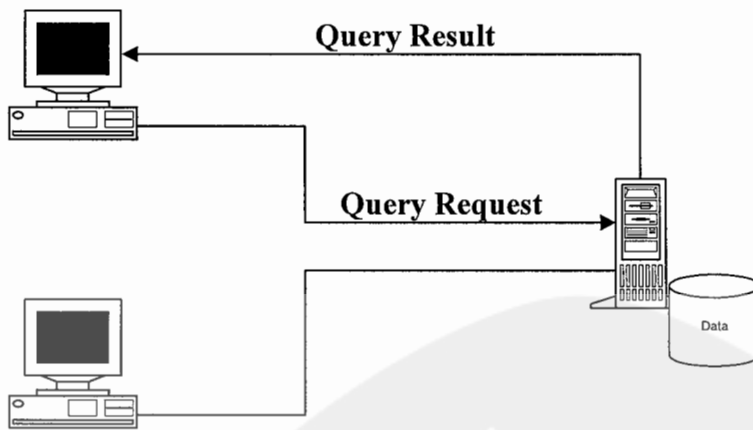
d. Tingkatan Komunikasi

Integrasi aplikasi yang dilakukan pada tingkatan sebelumnya juga terkait dengan integrasi aplikasi pada tingkatan komunikasi karena jika aplikasi tidak dapat berkomunikasi satu sama lain, maka tidak dapat diintegrasikan. Hal ini dapat terjadi karena tidak semua aplikasi dirancang untuk berkomunikasi dengan cara yang sama, ada aplikasi yang berkomunikasi secara sinkron, ada yang asinkron, ada yang menggunakan file transfer, dan sebagainya.

Pada saat akan melakukan integrasi aplikasi, selain memperhatikan tingkatan integrasi aplikasi, ada beberapa hal lain yang juga perlu diperhatikan, yaitu arsitektur dan tipe integrasi aplikasi. Beberapa arsitektur integrasi yang umum digunakan meliputi *Point to point Integration*, *Hub and Spoke Integration*, *Enterprise Message Bus Integration*, dan *Enterprise Service Bus Integration* (Binildas, 2008). Sedangkan tipe-tipe integrasi terdiri dari *Information Portals*, *Data Replication*, *Shared Business Functions*, *Service-Oriented Architectures*, *Distributed Business Processes*, dan *Business-to-Business Integration* (Hohpe, 2005).

### 2.3 Three-tier Client-server Architecture

*Client-server* merupakan sebuah model komputasi dimana sebuah aplikasi mengakses informasi dari aplikasi lainnya, bisa pada komputer yang sama atau pada komputer lain. Bagian *client* dari aplikasi tersebut dibangun untuk interaksi pengguna, sementara bagian *server* dibangun untuk menangani beberapa permintaan *client*. Model ini dapat diimplementasikan dalam lingkungan perangkat keras dan perangkat lunak yang berbeda-beda. Proses *client* dan *server* dapat dijalankan pada komputer yang sama ataupun berbeda, bahkan sebuah proses server dapat meminta layanan ke *server* lain.



**Gambar 2.4. Arsitektur two-tier**

Dalam *two-tier system* terdapat program *client* dan program *server*. Perbedaan utama dari keduanya adalah bahwa *server* merespon permintaan dari banyak *client* yang berbeda, sementara *client* biasanya berinisiatif melakukan permintaan informasi dari *server*. Dalam arsitektur ini terdapat dua pilihan pembangunan aplikasinya, *fat client* dan *thin client* (*fat server*), tergantung fungsionalitas yang berada di *client*. Model *fat client-thin server* merupakan sebuah model dimana hampir semua fungsionalitas dan *business logic* berada pada *client*. Sebaliknya model *thin client-fat server* menunjukkan hampir semua fungsionalitas dan *business logic* berada di *server*.

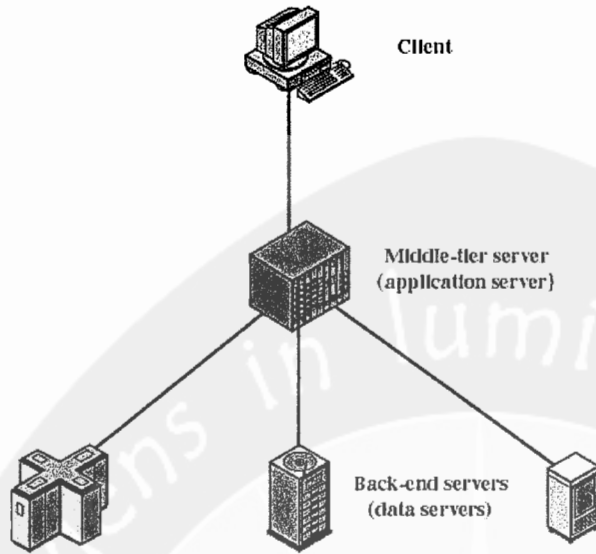
Arsitektur *three-tier* merupakan sebuah perbaikan dari *two-tier architecture*. Aliran informasi pada dasarnya tetap linear: sebuah permintaan (*request*) datang dari *client* ke *server*; *server* tersebut mengambil atau menyimpan data dari/ke basisdata; basisdata tersebut mengembalikan informasi ke server; server tersebut mengembalikan informasi ke *client*. Ada tiga bagian penting yang paling mendasar pada arsitektur ini, seperti terlihat pada Gambar 2.2 dan 2.3, adanya lapisan-lapisan yang menunjukkan bahwa arsitektur tersebut *3-tier*, yaitu :

- a. *User tier* (*presentation layer, user services*).
- b. *Business tier* (*business logic layer, business services*).
- c. *Data tier* (*data layer, data services*).

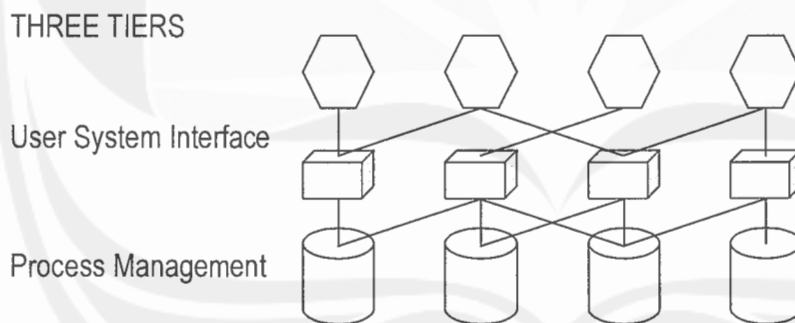
Perbedaan yang paling menonjol dari *2-tier* dengan *3-tier* yaitu adanya *middle tier* pada arsitektur *3-tier* yaitu *business logic layer*. Arsitektur *3-tier* muncul karena adanya keterbatasan pada arsitektur *2-tier*, dimana jika ada perubahan fungsi suatu komponen pada satu sisi mempengaruhi kedua sisi yaitu *client* dan *server*. Hal ini tidak terjadi pada sistem



3-tier. Arsitektur *3-tier* digunakan ketika diperlukan suatu rancangan *client-server* yang efektif, dimana meningkatkan kinerja, fleksibilitas, kemudahan perawatan, kemampuan untuk dapat digunakan ulang, dan skalabilitas.



Gambar 2.5. Three-tier client-server architecture.



Gambar 2.6. Konsep three-tier [SAD02].

Arsitektur *3-tier* memiliki komponen terdepan yang bertanggung jawab dalam penyediaan antarmuka ke pengguna, dan komponen paling belakang adalah *server* basisdata itu sendiri. Komponen *middle tier* memperbolehkan pengguna untuk berbagi dan mengontrol *business logic layer* dengan mengisolasi komponen-komponen yang ada didalam *middle tier* tersebut dari aplikasi yang sesungguhnya dihadapi pengguna. Dengan *middle tier* tersebut maka *business logic* dan *rules* dieksekusi dan dapat mencakup hingga

banyak *user*, lebih banyak daripada menggunakan *2-tier*, dengan menyediakan fungsi-fungsi untuk antrian, eksekusi aplikasi, maupun akses basisdata. Sistem pada *client* berinteraksi dengan *middle tier* melalui sebuah protokol yang sudah menjadi standar seperti misalnya HTTP, RPC, ataupun *middleware* seperti CORBA, DCOM, atau JavaBean. Dan *middle tier* tersebut berinteraksi dengan *database server* melalui protokol *database* standar seperti SQL, ODBC, dan JDBC. *Middle tier* merupakan lapisan yang paling penting dan menentukan adanya komunikasi antara *client* dengan *server*. Pada *middle tier* ini segala fungsi-fungsi yang dibutuhkan untuk komunikasi antara *client* dan *server* muncul. Adanya istilah *fat server* maupun *fat client* merupakan pengertian bahwa fungsi-fungsi tersebut cenderung lebih banyak menempel pada *client* atau pada *server*.

Keuntungan dari arsitektur *3-tier* yaitu :

- (1) Perubahan pada antarmuka pengguna atau *application logic* tidak saling mempengaruhi satu sama lain, membuat aplikasi tersebut mudah berevolusi untuk memenuhi kebutuhan baru.
- (2) *Network bottleneck* diminimalkan karena *application layer* tidak mentransmisikan data ekstra ke *client*, hanya data yang diperlukan untuk menangani suatu *task*.
- (3) Ketika *business logic* perlu diubah, hanya *server* yang perlu diubah. Dalam *two-tier architecture* setiap *client* perlu dimodifikasi.
- (4) *Client* terisolasi dari basisdata dan operasi jaringan. *Client* tersebut dapat mengakses data dengan mudah dan cepat tanpa harus mengetahui lokasi penyimpanan data atau jumlah *server* yang ada dalam sistem.
- (5) Koneksi basisdata dapat di-*pool* dan kemudian dibagi-pakai oleh beberapa pengguna, yang akan sangat mengurangi biaya sehubungan dengan *per-user licensing*.
- (6) Organisasi memiliki basisdata yang independen, karena *data layer* ditulis menggunakan SQL standar yang memang bebas *platform*. Perusahaan tersebut tidak terikat pada *vendor-specific stored procedures*.
- (7) *Application layer* ditulis dalam bahasa generasi ketiga atau keempat standar, yang telah biasa digunakan para pemrogram dari perusahaan tersebut.

## **BAB 3. ANALISIS DAN PERANCANGAN KEBUTUHAN INTEGRASI**

Bab ini berisi analisis dan perancangan kebutuhan integrasi SIATMA dengan Situs Kuliah Universitas Atma Jaya Yogyakarta.

### **3.1 Analisis Kebutuhan Integrasi**

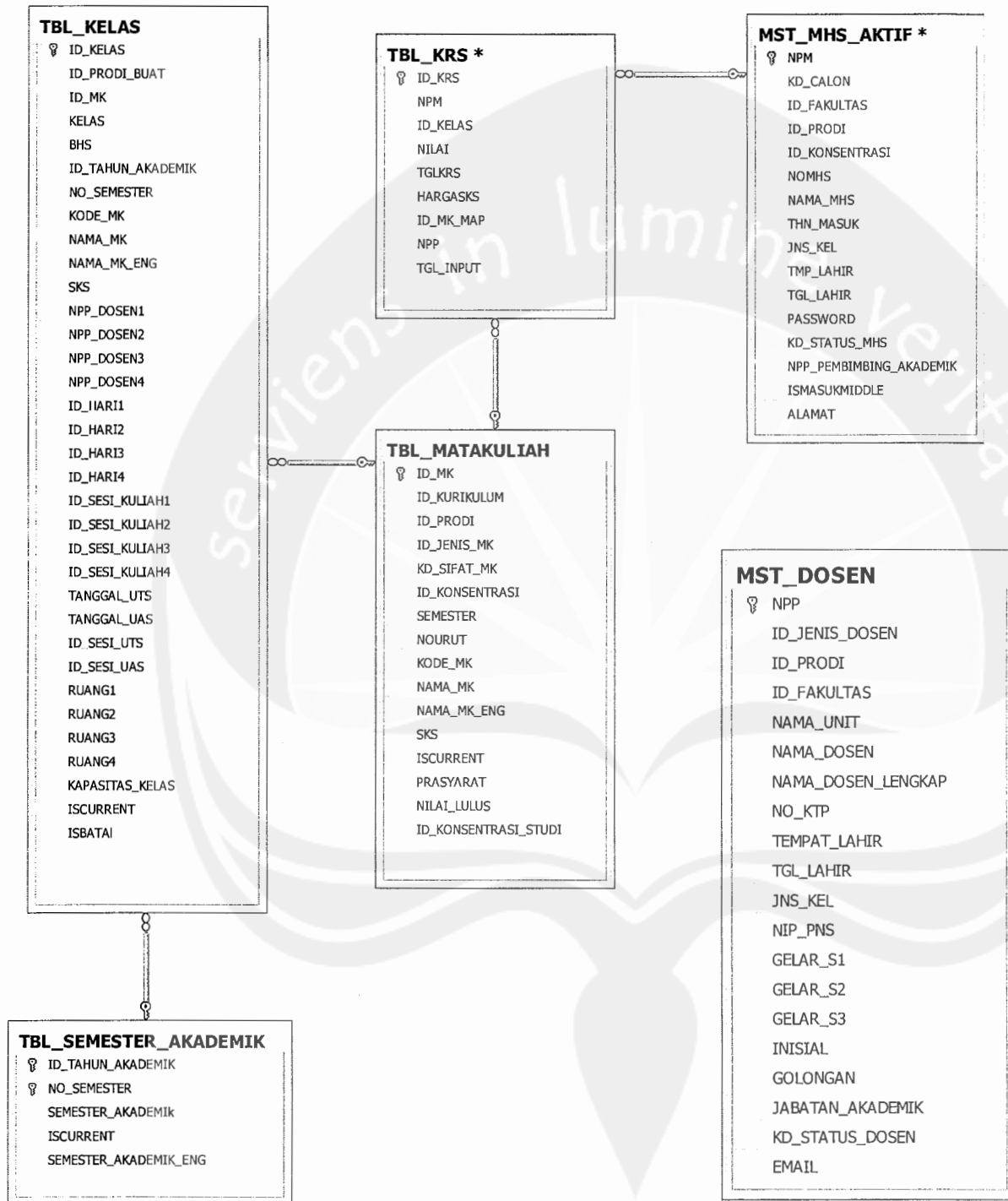
Saat ini, SIATMA dan Situs Kuliah, masing-masing memiliki basisdata sendiri-sendiri yang platform dan struktur basisdatanya berbeda. SIATMA menggunakan Microsoft SQL Server sebagai platform database management system (DBMS)-nya. Sedangkan Situs Kuliah menggunakan MySQL sebagai platform DBMS-nya. Meski platform dan struktur basisdatanya berbeda, sebenarnya ada beberapa data yang ada di kedua sistem tersebut yang saling terkait, seperti data mata kuliah, data pengajar, dan data peserta kuliah (mahasiswa/i).

Proses yang terjadi saat ini, pengelola Situs Kuliah memasukkan data-data mata kuliah dan data-data pengajar ke dalam Situs Kuliah secara manual menggunakan antarmuka sistem yang disediakan di Situs Kuliah. Data-data peserta mata kuliah tidak dimasukkan ke Situs Kuliah oleh pengelola Situs Kuliah melainkan mahasiswa diminta untuk melakukan proses enrolment sendiri-sendiri ke mata kuliah yang diambil di Situs Kuliah. Hal ini tentu saja dapat menimbulkan permasalahan bagi pengelola Situs Kuliah maupun Mahasiswa.

Hal tersebut di atas menunjukkan bahwa tingkatan integrasi aplikasi yang tepat untuk kasus tersebut adalah integrasi pada tingkatan data. Cara yang dapat dilakukan untuk melakukan integrasi pada tingkatan data adalah dengan menambahkan logic yang memungkinkan setiap aplikasi untuk memahami data yang datang dari aplikasi lainnya atau menambahkan logic yang memungkinkan setiap aplikasi menginterpretasikan data yang dihasilkan aplikasi ke dalam format tertentu dan diinterpretasikan ke dalam format yang dipahami aplikasi.

### 3.1.1 Spesifikasi Struktur Data SIATMA

Berdasarkan spesifikasi struktur data pada SIATMA, ada beberapa data yang menjadi kandidat untuk digunakan dalam integrasi pada tingkatan data, yaitu data mata kuliah, data dosen, data mahasiswa, data kelas, data KRS, dan data semester akademik.



### 3.1.2 Spesifikasi Struktur Data Situs Kuliah

Berdasarkan spesifikasi struktur data pada Situs Kuliah, ada beberapa data yang menjadi kandidat untuk digunakan dalam integrasi pada tingkatan data, yaitu data kuliah\_course, kuliah\_course\_categories, kuliah\_course\_modules, kuliah\_course\_sections, kuliah\_course\_display, kuliah\_course\_meta, kuliah\_course\_request, kuliah\_course\_allowed\_modules dan kuliah\_user.



Gambar 3.2. Spesifikasi Struktur Data SITUS KULIAH

### 3.1.3 Mapping Struktur Data SIATMA dan Situs Kuliah

Setelah kandidat spesifikasi struktur data pada SIATMA dan Situs Kuliah diidentifikasi, maka langkah berikutnya adalah melakukan mapping data dari SIATMA ke Situs Kuliah. Hasil dari proses mapping ini nantinya akan digunakan untuk membuat struktur file yang akan digenerate oleh SIATMA2KULIAH.

Tabel 3.1. Mapping Struktur Data SIATMA dan Situs Kuliah

Struktur Data SIATMA	Struktur Data Situs Kuliah
TBL_KELAS.NAMA_MK + TBL_KELAS.KELAS	fullname
TBL_KELAS.KODE_MK + TBL_KELAS.KELAS	shortname
TBL_SEMESTER_AKADEMIK.SEMESTER_AKADEMIK	category
-	sortorder
-	idnumber
TBL_KELAS.NAMA_MK	summary
-	format
-	showgrades
-	newsitems
-	teacher
-	teachers
-	student
-	students
-	startdate
-	numsections
-	maxbytes
-	visible
-	groupmode
-	timecreated
-	timemodified
TBL_KELAS.KODE_MK + TBL_KELAS.KELAS	password
-	enrolperiod
-	groupmodeforce
-	metacourse

-	lang
-	theme
-	cost
-	showreports
-	guest
-	enrollable
-	enrolstartdate
-	enrolenddate
-	notifystudents
-	expirynotify
-	expirythreshold
-	teacher1_role
MST_DOSEN.EMAIL	teacher1_account

Tabel di atas menunjukkan bahwa masih ada beberapa field dari struktur data Situs Kuliah yang tidak bisa diambil dari struktur data SIATMA. Field-field tersebut ada yang boleh kosong dan ada yang diisi dengan nilai default sesuai dengan yang diinginkan oleh pengelola Situs Kuliah. Nilai-nilai default yang sebaiknya di-assign ke field-field tersebut dapat dilihat pada Tabel berikut.

Tabel 3.2. Nilai Default pada Struktur Data Situs Kuliah

Field Struktur Data Situs Kuliah	Nilai Default
sortorder	1
summary	summary
format	topic
showgrades	0
newsitems	0
teacher	Owner
teachers	Owners
student	Visitor
students	Visitors
startdate	(ditetapkan sesuai dengan tanggal mulai semester)

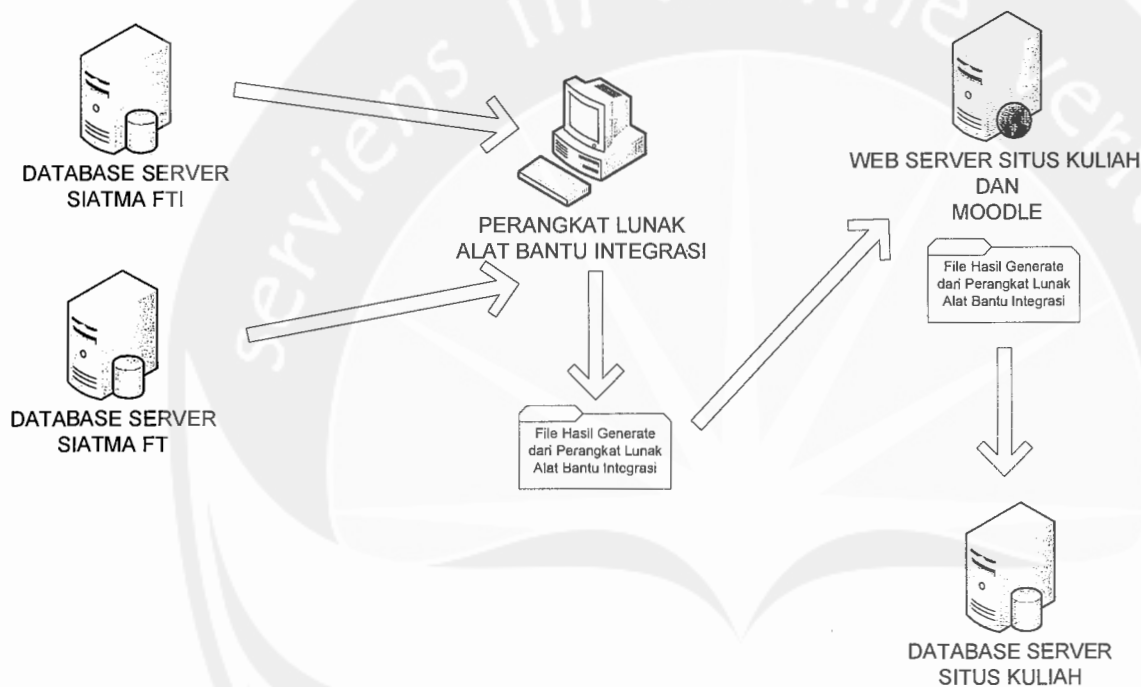
numsections	14
maxbytes	15728640
visible	0
groupmode	0
timecreated	(mengambil tanggal dari sistem)
timemodified	(mengambil tanggal dari sistem)
enrolperiod	0
groupmodeforce	0
metacourse	0
lang	-
theme	-
cost	-
showreports	0
guest	2
enrollable	1
enrolstartdate	(ditetapkan sesuai dengan tanggal mulai semester)
enrolenddate	(ditetapkan sesuai dengan tanggal akhir semester)
notifystudents	0
expirynotify	0
expirythreshold	10
teacher1_role	editingteacher



### 3.2 Perancangan Kebutuhan Integrasi

Analisis kebutuhan integrasi menunjukkan bahwa integrasi terjadi pada tingkat data sehingga yang dibutuhkan untuk proses integrasi adalah perangkat lunak alat bantu proses integrasi yang akan menjembatani proses integrasi (bridging data). Perangkat lunak alat bantu ini berperan sebagai logic yang memungkinkan setiap aplikasi untuk memahami data yang datang dari aplikasi lainnya atau menambahkan logic yang memungkinkan setiap aplikasi menginterpretasikan data yang dihasilkan aplikasi ke dalam format tertentu dan diinterpretasi ke dalam format yang dipahami aplikasi.

Berdasarkan kebutuhan tersebut, maka rancangan integrasi SIATMA dan Situs Kuliah adalah sebagai berikut:



**Gambar 3.3. Rancangan Arsitektur Integrasi Aplikasi**

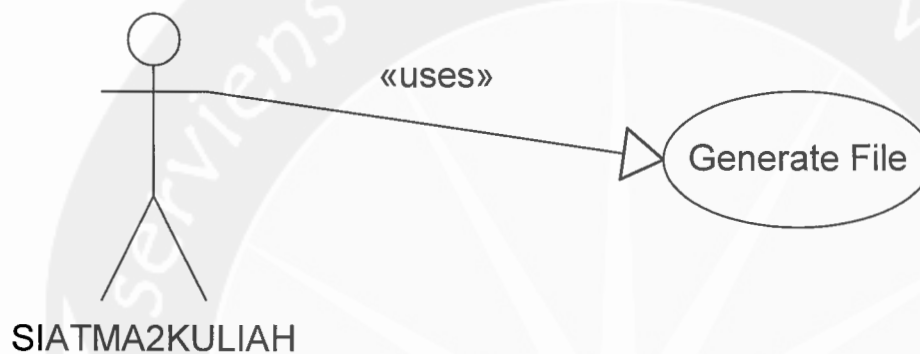
Peranglat lunak alat bantu integrasi akan mengambil data-data yang tersedia pada SIATMA sesuai dengan yang dibutuhkan oleh Situs Kuliah. Data yang diambil kemudian digunakan untuk mengenerate file (comma separated value) dengan format yang sesuai dengan yang dibutuhkan pada Situs Kuliah. File tersebut kemudian diupload ke Web Server untuk diproses dan disimpan di basis data situs kuliah.

## BAB 4. ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi spesifikasi kebutuhan dan perancangan perangkat lunak alat bantu integrasi SIATMA dengan Situs Kuliah Universitas Atma Jaya Yogyakarta (selanjutnya disebut SIATMA2KULIAH).

### 4.1 Spesifikasi Kebutuhan Fungsional

Berdasarkan analisis, kebutuhan fungsionalitas dari SIATMA2KULIAH ditunjukkan dengan diagram *use case* pada gambar 3.2., dan deskripsi rincinya ditunjukkan dengan *use case glossary* pada tabel 3.1.



Gambar 4.1. Diagram Use Case SIATMA2KULIAH

Tabel 4.1. Use Case Glossary SIATMA2KULIAH

<i>Use case Name</i>	<i>Use case Description</i>
Generate File	<i>Use case ini menyediakan fungsionalitas untuk mengenerate file (comma separated value) berisi data-data dari SIATMA yang akan digunakan di Situs Kuliah.</i>

### 4.2 Spesifikasi Kebutuhan Data

Dari analisis, data yang dibutuhkan untuk disimpan sebagai data *persistent* dalam bentuk file *comma separated value* adalah *fullname*, *shortname*, *category*, *sortorder*, *idnumber*, *summary*, *format*, *showgrades*, *newsitems*, *teacher*, *teachers*, *student*, *students*, *startdate*, *numsections*, *maxbytes*, *visible*, *groupmode*, *timecreated*, *timemodified*,

password, enrolperiod, groupmodeforce, metacourse, lang, theme, cost, showreports, guest, enrollable, enrolstartdate, enrolenddate, notifystudents, expirynotify, expirythreshold, teacher1\_role, teacher1\_account. Spesifikasi kebutuhan data dari aplikasi SIATMA2KULIAH dapat dilihat pada Tabel 4.2.

Tabel 4.2. Tabel Spesifikasi Kebutuhan Data

Field Struktur Data Situs Kuliah
fullname
shortname
category
sortorder
idnumber
summary
format
showgrades
newsitems
teacher
teachers
student
students
startdate
numsections
maxbytes
visible
groupmode
timecreated
timemodified
password
enrolperiod
groupmodeforce
metacourse
lang

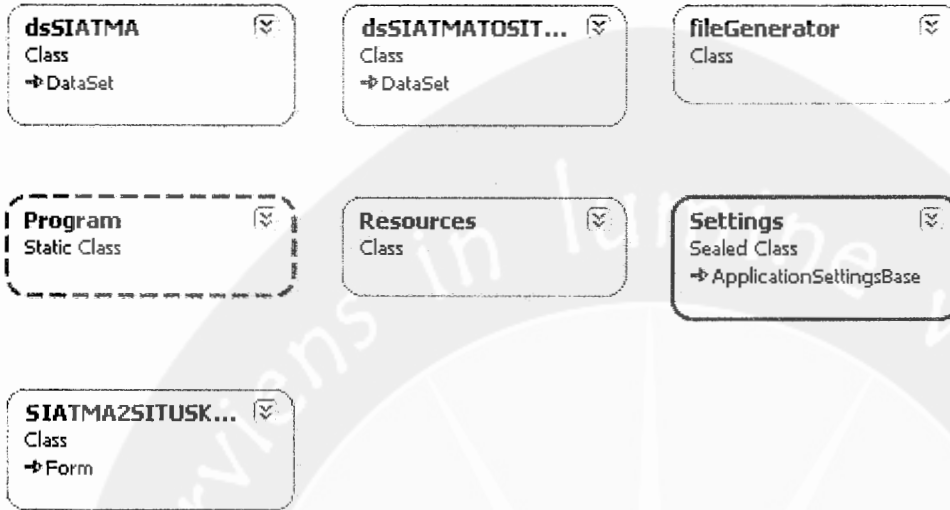
theme
cost
showreports
guest
enrollable
enrolstartdate
enrolenddate
notifystudents
expirynotify
expirythreshold
teacher1_role
teacher1_account

### 4.3 Spesifikasi Kebutuhan Antarmuka Eksternal

Kebutuhan antar muka eksternal pada perangkat lunak SIATMA2KULIAH meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi. Untuk antarmuka pemakai, pengguna akan berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form (SIATMA2KULIAH) dan page yang memiliki berbagai elemen kontrol untuk berinteraksi dengan aplikasi. Kebutuhan perangkat keras untuk menjalankan aplikasi ini berupa PC. Sedangkan kebutuhan perangkat lunak meliputi Sistem Operasi Windows, Microsoft SQL Server 2005, Web Browser, Moodle, Moodle Plugins for Course Bulk Upload dan NET Framework 2.0. Kebutuhan antarmuka komunikasi adalah protokol TCP/IP dan HTTP.

## 4.4 Perancangan Fungsional

Fungsionalitas yang dideskripsikan dalam bentuk *use case* pada bab sebelumnya, selanjutnya direalisasi dalam bentuk kelas-kelas yang mengimplementasikan fungsionalitas tersebut.



Gambar 4.2. Diagram Kelas SIATMA2KULIAH

## 4.5 Perancangan Data

Selanjutnya, berdasarkan analisis kebutuhan data yang telah dilakukan sebelumnya, maka dilakukan perancangan tabel data sebagai berikut:

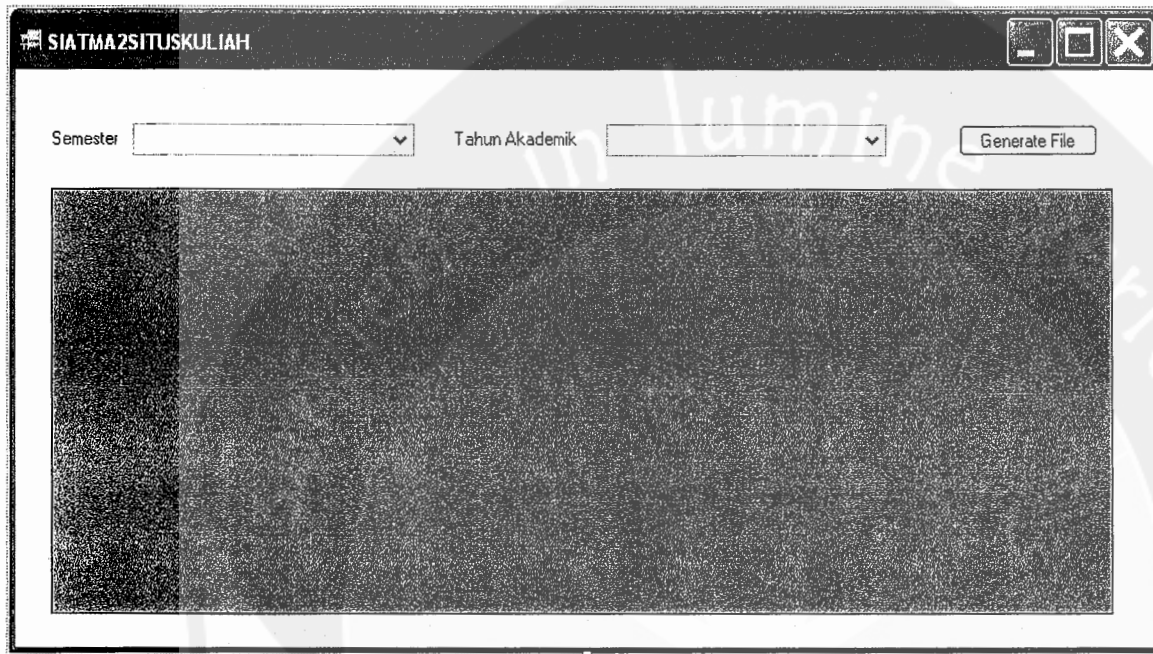
Tabel 4.3. Dekomposisi Data File

Field Struktur Data Situs Kuliah
fullname
shortname
category
sortorder
idnumber
summary
format
showgrades

newsitems
teacher
teachers
student
students
startdate
numsections
maxbytes
visible
groupmode
timecreated
timemodified
password
enrolperiod
groupmodeforce
metacourse
lang
theme
cost
showreports
guest
enrollable
enrolstartdate
enrolenddate
notifystudents
expirynotify
expirythreshold
teacher1_role
teacher1_account

#### 4.6 Perancangan Arsitektur Navigasi Antar Muka Pengguna

Berikutnya akan dibahas perancangan arsitektur navigasi antar muka pengguna aplikasi. Arsitektur navigasi antar muka pengguna aplikasi SIATMA2KULIAH terdiri dari form utama yang digunakan untuk mengenerate file berdasarkan pilihan Semester dan Tahun Akademik. Rancangan antar muka pengguna aplikasi SIATMA2KULIAH dapat dilihat pada Gambar berikut.



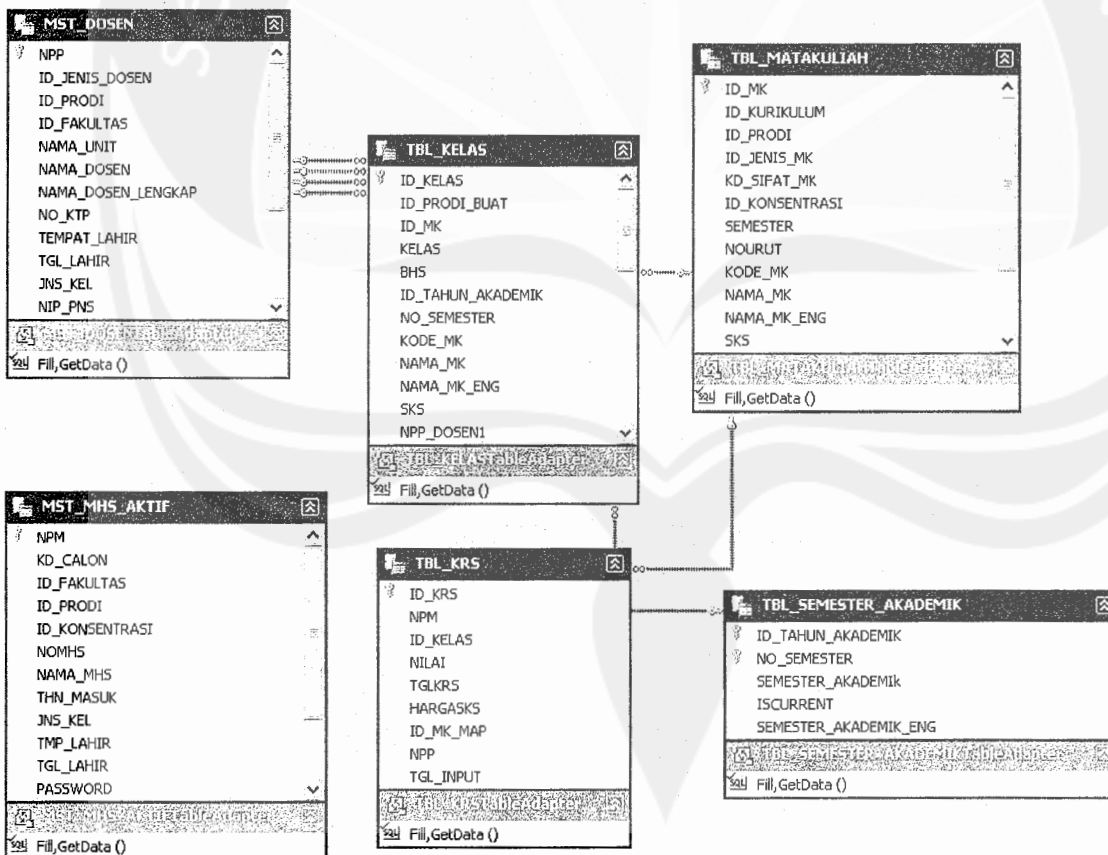
**Gambar 4.3. Rancangan Antar Muka Pengguna SIATMA2KULIAH: Form Utama**

## BAB 5. PROSES INTEGRASI SISTEM

Bab ini berisi pembahasan mengenai proses integrasi SIATMA dengan Situs Kuliah menggunakan perangkat lunak alat bantu integrasi SIATMA dengan Situs Kuliah Universitas Atma Jaya Yogyakarta (SIATMA2KULIAH).

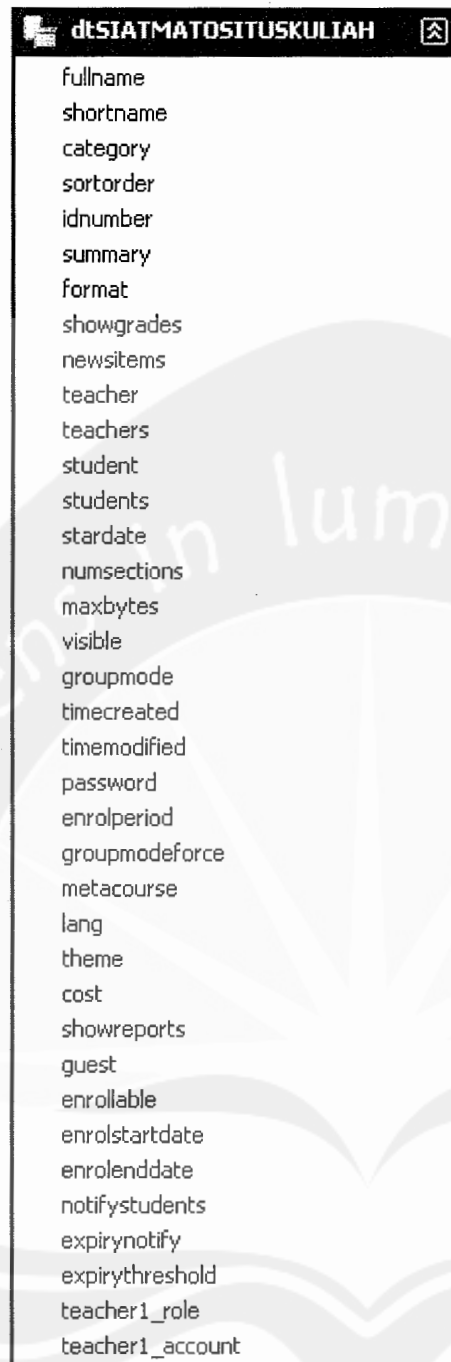
### 5.1 Proses Generate File

Proses *generate file* yang akan digunakan untuk keperluan integrasi SIATMA dan Situs Kuliah pada tingkatan data akan dilakukan dengan membaca data-data yang diperlukan dari SIATMA untuk dimasukkan ke dalam DataSet yang formatnya sudah sesuai dengan struktur file yang akan di-generate (Gambar 5.2). Data dari SIATMA digunakan untuk mengisi beberapa field saja seperti yang sudah dijelaskan pada bab sebelumnya (Tabel 3.1.). DataSet SIATMA diisi terlebih dahulu dengan filter Semester Akademik dan Tahun Akademik, sehingga data yang di-load ke DataSet tidak semuanya untuk mempercepat kinerja proses *generate file*.



Gambar 5.1. DataSet SIATMA





**Gambar 4.2. DataSet Struktur File**

Selain mengambil data dari SIATMA, proses yang terjadi pada saat generate file adalah mengisi DataSet Struktur File dengan nilai-nilai default sesuai dengan yang sudah dijelaskan pada bab sebelumnya (Tabel 3.2.). Setelah proses generate file selesai, maka akan dihasilkan file berformat comma separated value (.csv) yang nantinya akan digunakan pada proses selanjutnya.

## 5.2 Proses Bulk Course Upload

Proses *bulk course upload* dilakukan dengan mengakses halaman Administrator Situs Kuliah. Ada beberapa hal yang harus diperhatikan sebelum melakukan proses ini, yaitu data-data teacher (kuliah\_course\_users) dan kategori mata kuliah (kuliah\_course\_categories) harus disiapkan terlebih dahulu. Setelah semuanya siap, maka proses bulk course upload dapat dilakukan melalui halaman pengelolaan Mata kuliah di Situs Kuliah. Proses *bulk course upload* dilakukan dengan memilih file *comma separated value* yang sudah dihasilkan sebelumnya.

The screenshot displays the 'Add/edit courses' interface. On the left, there is a 'Site Administration' sidebar with various options like Notifications, Users, Courses, Enrolments, etc. The main content area is titled 'Course categories' and contains a table with the following data:

Course categories	Courses	Edit
Fakultas Teknologi Industri	0	↵ ✕ ⓧ
Program Studi Teknik Informatika	86	↵ ✕ ⓧ
Program Studi Teknik Industri	87	↵ ✕ ⓧ
Program Studi Teknik Industri Kelas Internasional	87	↵ ✕ ⓧ
Program Studi Teknik Industri Kelas ATMI	2	↵ ✕ ⓧ
Fakultas Ekonomi	1	↵ ✕ ⓧ
Program Studi Manajemen	69	↵ ✕ ⓧ
Semester Pendek 2010	2	↵ ✕ ⓧ
Semester Gasal 2010/2011	69	↵ ✕ ⓧ
Program Studi Akuntansi	39	↵ ✕ ⓧ
Semester I Pendek UAJY	26	↵ ✕ ⓧ
Semester Gasal 2010/2011	51	↵ ✕ ⓧ
Program Studi Ilmu Ekonomi	41	↵ ✕ ⓧ
Semester Gasal 2010/2011	49	↵ ✕ ⓧ
International Business Management Program (IBMP)	0	↵ ✕ ⓧ

Below the table, there is a 'Move category to:' dropdown menu with options like 'Top', 'Fakultas Teknologi Industri', 'Fakultas Ekonomi', etc.

Gambar 4.3. DataSet Struktur File

## BAB 6. KESIMPULAN

Integrasi sistem membutuhkan proses analisis kebutuhan dan perancangan integrasi untuk menentukan tingkatan integrasi yang diperlukan. Pada kasus integrasi sistem SIATMA dan Situs Kuliah di Universitas Atma Jaya Yogyakarta, tingkatan integrasi yang dipilih adalah pada data. Integrasi pada tingkatan data dilakukan dengan menambahkan logic yang memungkinkan setiap aplikasi untuk memahami data yang datang dari aplikasi lainnya atau menambahkan logic yang memungkinkan setiap aplikasi menginterpretasikan data yang dihasilkan aplikasi ke dalam format tertentu dan diinterpretasiokan ke dalam format yang dipahami aplikasi.

Penelitian-penelitian selanjutnya dapat difokuskan untuk mengembangkan modul pada SIATMA sehingga proses integrasi dapat dilakukan secara langsung. Proses integrasi yang dilakukan pada penelitian ini masih mengharuskan pengguna untuk mengakses Situs Kuliah dan menggunakan Moodle Plugins Bulk Course Upload. Jika modul pada SIATMA dapat ditambahkan, maka ketika proses pembuatan penawaran kelas secara otomatis juga akan melakukan update pada database Situs Kuliah.



## DAFTAR PUSTAKA

**Binildas C.A., Service Oriented Java Business Integration, PACKT Publishing, 2008.**

**Hohpe, Gregor; Woolf, Bobby; Enterprise Integration Patterns: Designing, Building and Deploying Messaging Solutions, Addison Wesley, 2005.**

**Juric, Matjaz B.; Loganathan, Ramesh; Sarang, Poornachandra; Jennings, Frank; SOA Approach to Integration: XML, Web services, ESB, and BPEL in Real-world SOA Projects, Packt Publishing, 2007.**

**Microsoft Corporation, Guidelines for Application Integration: Patterns and Practices, Microsoft Corporation, 2003.**

**Sadoski, D., S. Comella-Dorda, diakses 18/12/2010, Three Tier Software Architectures, <http://www.sei.cmu.edu>.**