

Sinkronisasi Jadwal Perkuliahan pada Aplikasi Android menggunakan Teknologi XML-RPC

(Studi Kasus di STMIK AMIKOM Yogyakarta)

Arif Dwi Laksito

Jurusan Teknik Informatika STMIK AMIKOM Yogyakarta
Jl. Ring Road Utara Condong Catur Depok Sleman Yogyakarta
arif.laksito@amikom.ac.id

Abstrak—Saat ini perkembangan teknologi perangkat bergerak (*mobile device*) cukup pesat, sehingga dengan teknologi tersebut memungkinkan pertukaran informasi dari aplikasi web dan aplikasi di *mobile*. Dengan memanfaatkan informasi jadwal perkuliahan yang telah ada di STMIK AMIKOM Yogyakarta maka dibangun aplikasi *mobile* berbasis android yang dapat menampilkan data tersebut dan dapat dimanfaatkan sebagai pengingat(*reminder*). Dengan adanya Teknologi *web service* memungkinkan komunikasi antar aplikasi dengan menggunakan XML(*Extensible Markup Language*) untuk melakukan pertukaran data. Dengan adanya *web service* maka pertukaran data jadwal kuliah dosen dan mahasiswa pada *server web* di STMIK AMIKOM Yogyakarta dan aplikasi *mobile* dapat dilakukan. Penggunaan teknologi XML-RPC dipilih karena mudah dalam implementasi pertukaran data di bahasa pemrograman PHP pada web dengan Java di Android serta sudah tersedia *library* yang langsung bisa digunakan. Untuk memudahkan sinkronisasi data ke *account* Google digunakan *iCalendar* sebagai format standard penanggalan di internet.

Kata kunci—*web service; xml-rpc; xml; android; iCalendar*

I. PENDAHULUAN

Informasi jadwal perkuliahan untuk mahasiswa dan dosen di STMIK AMIKOM Yogyakarta sudah bisa diakses di website www.amikom.ac.id mulai sekitar tahun 2006. Informasi yang tersaji tersebut sudah bersifat *private* untuk tiap mahasiswa dan dosen yaitu jadwal antara mahasiswa A dan mahasiswa B dapat berbeda sesuai dengan matakuliah yang diambil saat pengisian KRS, begitu juga pada jadwal dosen yang mengampu matakuliah tersebut berbeda dengan dosen yang lain.

Perkembangan teknologi perangkat bergerak (*mobile device*) cukup pesat, sehingga dengan teknologi tersebut memungkinkan pertukaran informasi dari aplikasi web dan aplikasi di *mobile*. Salah satu perkembangan teknologi di *smartphone* adalah sistem operasi *Android*. *Android* merupakan sistem operasi yang tergolong masih baru dan terus mengalami perkembangan. Sistem operasi ini bersifat

open source sehingga memudahkan para pengembang aplikasi untuk membangun aplikasi pada *platform Android*. Pertama kali sistem operasi ini dirilis oleh Google pada tahun 2009 dan sejak saat itu *Android* terus berkembang dengan pesat dan telah berhasil mendapat perhatian dari jutaan pengguna dan pengembang aplikasi *mobile phone*.

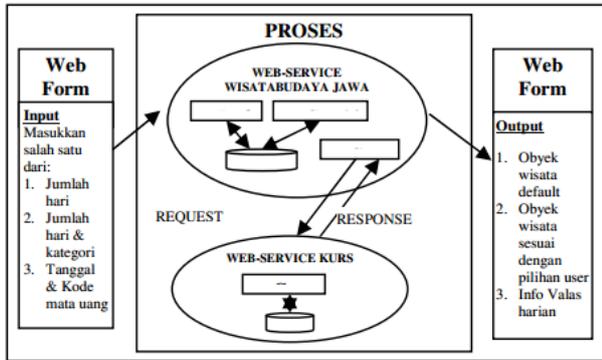
Teknologi *web service* memungkinkan komunikasi antar aplikasi dengan menggunakan XML(*Extensible Markup Language*) untuk melakukan pertukaran data. Menurut Menes *web service* yaitu teknologi yang memungkinkan bentuk akhir dari sebuah program atau aplikasi komputer adalah berupa sebuah *service* atau fungsi yang akan melakukan sebuah tugas atau proses yang spesifik [4].

Penelitian ini mengambil topik pembahasan mengenai *web service* dan mengimplementasikannya untuk sinkronisasi jadwal perkuliahan di STMIK AMIKOM Yogyakarta menggunakan format *iCalendar* yang dapat terintegrasi dengan *account* google dan sebagai *event calendar* di *Android mobile phone*.

II. TINJAUAN PUSTAKA

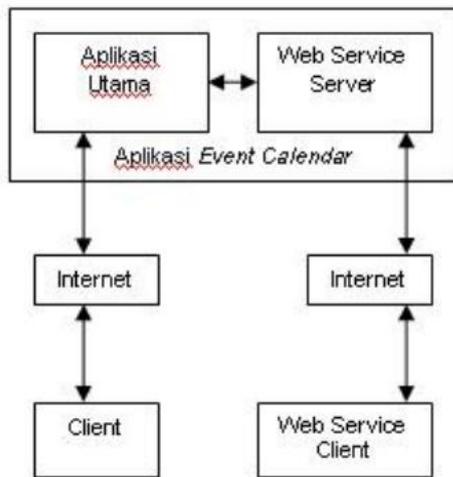
Sebelumnya telah dilakukan penelitian tentang *web service* oleh Tri Kuntoro Priyambodo dengan judul Implementasi *Web Service* untuk Pengembangan Sistem Layanan Wisata Terpadu [6] dan penelitian tentang event calendar dan *web service* oleh Vidyapati dengan judul Pembangunan Aplikasi *Web Event Calendar* dengan dukungan *Web Service* [8].

Penelitian oleh Tri Kuntoro Priyambodo menghasilkan suatu rekomendasi dari *web service* yaitu info wisata default dan info wisata sesuai pilihan user dan info valas harian berdasarkan input dari user yaitu jumlah hari, jumlah hari dan kategori, tanggal dan kode mata uang.



Gambar 1. Arsitektur sistem web service WisataBudayaJawa

Penelitian vidyapati menghasilkan suatu aplikasi *web event calendar* dan *web service* yang menyediakan fungsi-fungsi seperti: memberikan *event* pada bulan dan tahun tertentu, memberikan *event* sesuai dengan *id* tertentu dan memberikan *event* sesuai kategori *event* tertentu yang dapat digunakan oleh *web service client*.



Gambar 2. Arsitektur sistem web event calendar

A. Web Service

Menurut Kreger *web service* diartikan sebagai sebuah antar muka (*interface*) yang menggambarkan sekumpulan operasi-operasi yang dapat diakses melalui jaringan, misalnya internet, dalam bentuk pesan *XML* [3]. Sedangkan menurut Manes (2001), *web service* diartikan sebagai sepotong atau sebagian informasi atau proses yang dapat diakses oleh siapa saja, kapan saja dengan menggunakan piranti apa saja, tidak terikat dengan sistem operasi atau bahasa pemrograman yang digunakan [4].

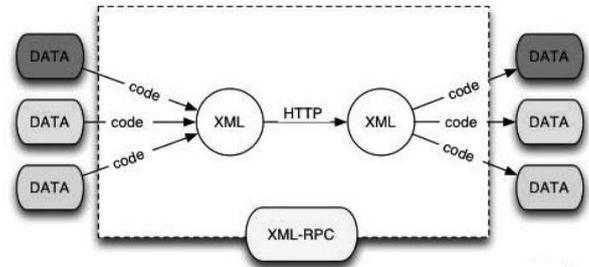
Web service merupakan *middleware* dalam suatu jaringan internet yang memungkinkan suatu sistem dapat saling berkomunikasi pada *platform* yang berbeda. *Web service* menggunakan antarmuka berupa *XML* yang dapat digunakan

untuk membawa operasi-operasi dan data untuk komunikasi antar sistem.

B. XML-RPC

XML-RPC adalah salah satu model *web service* yang bekerja pada protokol *RPC* dalam suatu jaringan internet. Sebuah message *XML-RPC* adalah *HTTP-POST request* berbasis *HTTP 1.0* dan *HTTP 1.1*.

XML-RPC menggunakan *XML* sebagai standarisasi pesan yang dikirim. Dengan penggunaan format tersebut, maka *platform client* dan server tidak dibatasi lagi oleh suatu bahasa pemrograman tertentu.



Gambar 3. Konsep dasar XML-RPC

Tujuan utama penggunaan *XML-RPC* adalah kemudahan dalam implementasi dan adaptasi di lingkungan pemrograman yang berbeda atau sistem operasi yang berbeda. *XML-RPC* digunakan untuk mencari cara yang efektif dan simple dalam hal permintaan dan penerimaan informasi [5].

C. XML

Menurut Walsh, *XML* merupakan sebuah *Markup Language* untuk dokumentasi terstruktur. Dokumen-dokumen terstruktur adalah dokumen-dokumen yang mempunyai isi/*content* (kata, gambar) serta indikasi yang menyatakan makna dari *content* tersebut [9].

Sebuah objek data dapat dikatakan sebagai dokumen *XML* jika memenuhi kriteria *well-formed*. Adapun kriteria *well-formed* adalah sebagai berikut [2]:

1. Satu dan hanya ada satu *root element* yang ada dalam suatu dokumen
2. Setiap elemen harus memiliki *start-tag* dan *end-tag*.
3. Setiap elemen harus *properly nested*.
4. Karakter pertama dari nama atribut harus berupa huruf alphabet atau *underscore*.
5. Suatu nama atribut hanya boleh muncul satu kali pada *start-tag* yang sama.

D. iCalendar (RFC2445)

RFC2445 berisi spesifikasi mengenai penjadwalan dan pengalenderan internet. RFC2445 menjadi standar bagi format

kalender yang ditransmisikan melalui internet dan dinamakan *iCalendar*. Standar ini telah banyak digunakan oleh beberapa aplikasi antara lain *Microsoft Outlook*, *iCal*, *Google Calendar*, *Mozilla Thunderbird*, dan lain-lain [1].

Terdapat beberapa tipe yang dapat digunakan pada *iCalendar*, yaitu [1]:

1. *vEvent*, mendeskripsikan bagian dari sebuah *event* yang dapat berisi *vAlarm* sebagai pengingat dalam suatu *event*.
2. *vTodo*, mendeskripsikan list pekerjaan yang sudah dilakukan atau belum.
3. *vJournal*, mendeskripsikan entry jurnal yang dilakukan secara rutin atau tidak sehingga berpengaruh terhadap *vFreeBusy*.
4. *vFreeBusy*, mendeskripsikan informasi dalam suatu kalender dimana tidak ada *event* atau kegiatan lain.
5. *vTimeZone*, mendeskripsikan *time zone*
6. *vAlarm*, mendeskripsikan *alarm*(pengingat) dalam *vEvent*.

E. Android

Android merupakan sistem operasi untuk *smartphone* yang dibuat oleh *google corporation* dengan memanfaatkan *Linux Kernel*. *Kernel* berperan sebagai *abstraction layer* antara *hardware* dan keseluruhan *software*. *Android* menyediakan *platform* terbuka bagi para pengembang untuk mengembangkan atau menciptakan aplikasi.

Pada versi 3.0 keatas perangkat selular yang digunakan akan berbentuk tablet yang memiliki ukuran yang lebih besar sehingga wadah untuk menampilkan aplikasinya akan user friendly bagi para penggunanya [7].

Beberapa fitur yang ada di *Android* adalah sebagai berikut [7]:

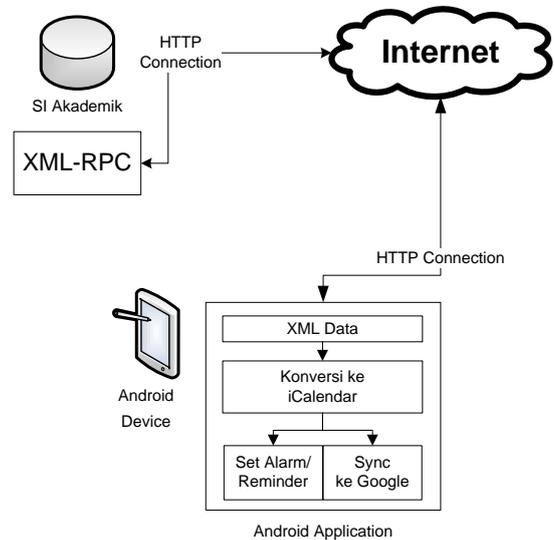
1. *Application Framework*, yang memungkinkan penggunaan dan penghapusan komponen yang tersedia.
2. *Dalvik Virtual Machine*, yaitu mesin virtual yang dioptimalkan untuk perangkat *mobile*.
3. *Graphic Library*, yang mendukung grafik 2D dan 3D berdasarkan *OpenGL Library*.
4. *Media Supported*, yang mendukung beberapa media seperti: audio, video, dan berbagai format gambar.
5. *Hardware Independent*, mendukung *GSM*, *Bluetooth*, *Wifi*, *Camera*, *GPS*, *Kompas*, *Accelerometer*, dll.

III. METODE PENELITIAN

Dalam penelitian ini dilakukan beberapa tahapan proses untuk menghasilkan aplikasi sinkronisasi jadwal perkuliahan untuk dosen dan mahasiswa yang dijelaskan sebagai berikut:

A. Arsitektur Sistem

Pemanfaat teknologi *mobile* dapat dilakukan untuk komunikasi dengan sistem berbasis web melalui koneksi data *wifi* atau jaringan paket data (*GPRS/3G/HSDPA*). Bentuk arsitektur sistem yang dikembangkan dapat dilihat pada gambar 4.



Gambar 4. Arsitektur Sistem

B. Analisis Proses

Secara umum proses terbagi menjadi 2 yaitu di bagian *server* dan di bagian *client*. Setiap proses memiliki peran masing-masing yang dijelaskan sebagai berikut:

1. Server

Dari SI(Sistem Informasi) di STMIK AMIKOM yang sudah berjalan dibuat *XML-RPC Server* menggunakan *PHP* dengan beberapa fungsi yang kemudian dapat di akses oleh *client*.

2. Client

Menggunakan *library xmlrpc Android* untuk mengakses fungsi dari server yang kemudian diproses menjadi *iCalendar* untuk dapat di set sebagai *Alarm* dan di sinkronisasikan dengan *Google Account*.

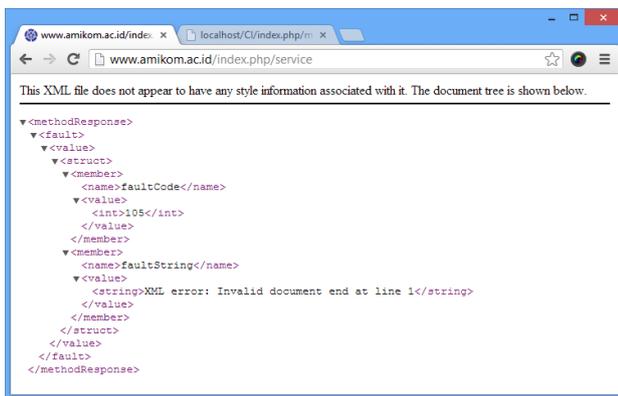
IV. HASIL DAN PEMBAHASAN

Berdasarkan arsitektur sistem dan analisis proses yang telah dirancang, maka dikembangkan beberapa fungsi di server yaitu:

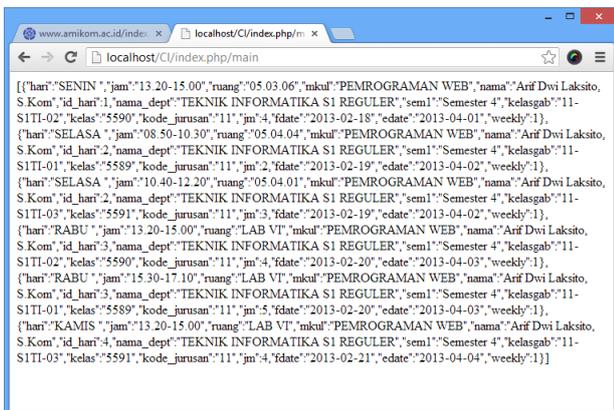
1. *loginmhs()*, digunakan untuk login mahasiswa dengan parameter npm (nomer pokok mahasiswa) dan password.
2. *logindosen()*, digunakan untuk login dosen dengan parameter nik dan password.
3. *jdwkuliah()*, mengambil data jadwal perkuliahan mahasiswa dengan parameter npm.
4. *jdwdosen()*, mengambil data jadwal mengajar dosen dengan parameter nik.
5. *jdwupdate()*, menampilkan versi jadwal terakhir di *server*, digunakan untuk update data di *client*.

Antarmuka tampilan di sisi *server* jika tidak ada *method* yang menggunakannya adalah seperti pada Gambar 5.

Salah satu format data yang dihasilkan dari fungsi *jdwdosen()* seperti pada Gambar 6.



Gambar 5. Antarmuka di Server



Gambar 6. Data dari fungsi *jdwdosen()*

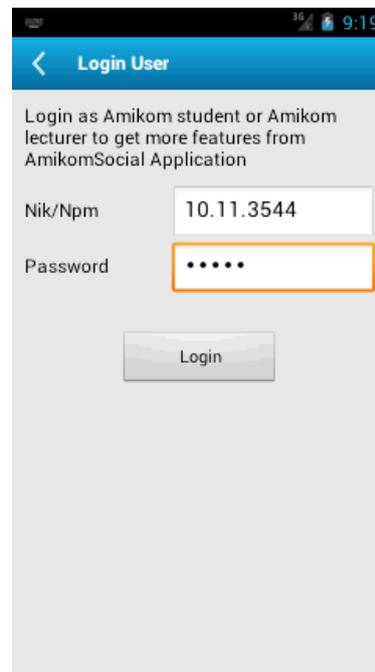
Pada sisi *client* pertama kali digunakan fungsi *loginmhs()* atau *logindosen()* dan jika sukses data-data mahasiswa atau dosen dari *server* akan di simpan di *database* pada aplikasi android.

Berdasarkan data pengguna di *database* akan digunakan fungsi *jdwkuliah()* untuk mahasiswa atau *jdwdosen()* untuk dosen, dan list jadwal tersebut akan di tampilkan di aplikasi android menggunakan *listview*. Data jadwal tersebut disesuaikan dengan jumlah pertemuan dosen dan mahasiswa serta jadwal untuk ujian tengah dan ujian akhir tiap mahasiswa.

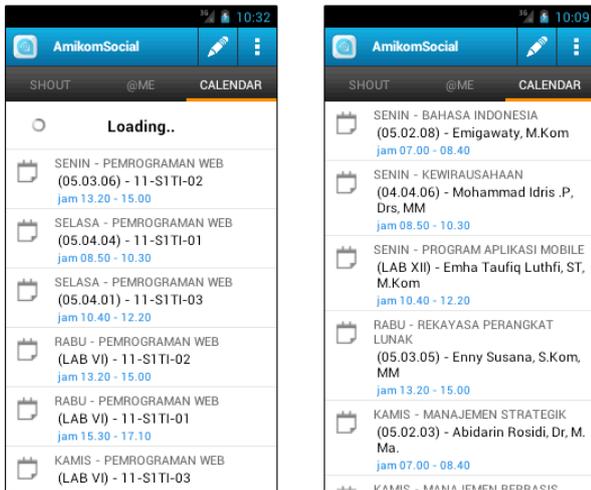
Secara umum pada *device Android* telah terdapat satu atau lebih *account Google*. Data jadwal yang telah tersimpan di *database* selanjutnya akan dikonversi menjadi *vEvent* yang selanjutnya disimpan di salah satu *account Google* yang dapat ditentukan oleh pengguna. Saat tersimpan sebagai *vEvent* secara otomatis data-data tersebut ter-sinkronisasi dengan *account Google* yang dapat di share ke user lain atau dapat ditampilkan di web. Pada *vEvent* tersebut telah di set *vAlarm* untuk pengingat yang secara default adalah 30 menit sebelum mulai perkuliahan.

Fungsi *jdwupdate()* digunakan untuk pengecekan data-data jadwal perkuliahan pada setiap pergantian semester, jika terdapat *update* versi jadwal terbaru, maka akan di gunakan fungsi *jdwkuliah()* atau *jdwdosen()* untuk mengambil data. Setiap terdapat *update* jadwal terbaru diperlukan sinkronisasi kembali untuk penambahan data *vEvent* dan *vAlarm* di *account Google*.

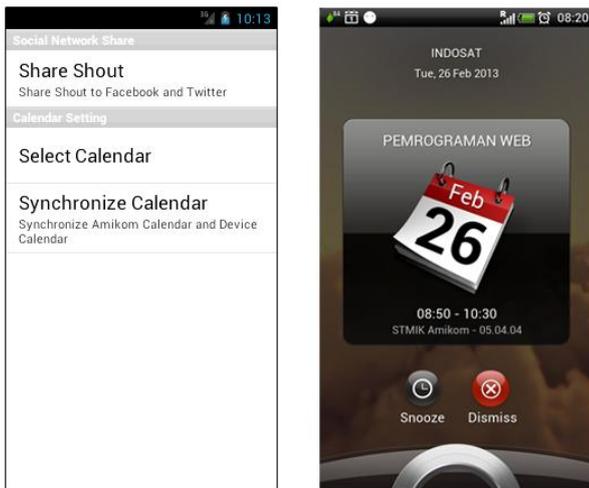
Antarmuka tampilan di *device Android* secara berurutan adalah sebagai berikut:



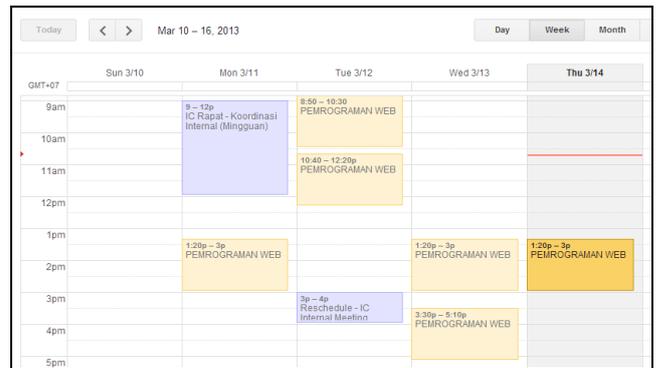
Gambar 7. Form login dosen atau mahasiswa



Gambar 8. Jadwal perkuliahan dan mengajar



Gambar 9. Sinkronisasi jadwal dan alarm pengingat



Gambar 10. Jadwal di account Google

V. KESIMPULAN DAN SARAN

Berdasarkan dari pembuatan aplikasi sinkronisasi jadwal perkuliahan tersebut, dapat diambil kesimpulan dan saran sebagai berikut:

A. Kesimpulan

Menghasilkan sebuah aplikasi sinkronisasi jadwal perkuliahan yang berjalan di Android smartphone yang dapat dijadikan pengingat jadwal kuliah untuk dosen dan mahasiswa di STMIK AMIKOM Yogyakarta. Proses pertukaran data dari server berbasis web ke aplikasi di Android berjalan dengan baik dan sinkronisasi jadwal bisa dijalankan di aplikasi ini.

B. Saran

Untuk pengembangan selanjutnya akan lebih baik jika lebih memperhatikan sisi keamanan dalam pertukaran data melalui teknologi *XML-RPC* dan untuk sinkronisasi aplikasi ini dapat dilakukan secara otomatis saat pergantian semester.

DAFTAR PUSTAKA

- [1] Dawson, F., Stenerson, D, 1998, "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", <http://tools.ietf.org/pdf/rfc2445>.
- [2] Jorgensen, David, 2002, "Developing .NET Web Services with XML", Syngress.
- [3] Kreger, H., 2001, "Web-services Conceptual Architecture (WSCA 1.0)", IBM Software Group, USA.
- [4] Manes, A.T., 2001, "Introduction to Web-Services", <http://www.systinet.com>.
- [5] Muhammad, Rizki, 2006, "Perancangan dan Pembuatan Sistem Informasi Pemasaran Barang Berbasis J2ME XML-RPC", Institut Teknologi Surabaya, Surabaya.
- [6] Priyambodo, Tri Kuntoro, 2005, "Implementasi Web-Service untuk Pengembangan Sistem Layanan Pariwisata Terpadu", Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005), Yogyakarta.
- [7] Priyanta, F, 2011, "Pemrograman Android untuk Pemula", Cerdas Pustaka Publisher, Jakarta,.
- [8] Vidyapati, 2008, "Pembangunan Aplikasi Web Event Calendar dengan dukungan Web Service", Institut Teknologi Bandung, Bandung.
- [9] Walsh, N., 1998, "A Technical Introduction to XML", Arbortext, In