

Membangun Web API dengan menggunakan JSON sebagai format serialisasi data

Muhammad Ghazali
Program Studi Teknik Informatika
Fakultas Teknik
Universitas Widyatama
<muhammad.ghazali@widyatama.ac.id>

25 Januari 2013

Daftar Isi

Latar Belakang dan Masalah	1
Rumusan Masalah	2
Batasan Masalah	2
Tujuan	2
Landasan Teori	3
Prosedur Penelitian	4
Jadwal Kerja	5
Usulan Pembimbing	5

Ringkasan

LayangLayang Mobile merupakan salah perusahaan yang bergerak di bidang *mobile application development*. Saat ini LayangLayang Mobile sedang mengembangkan sebuah produk bernama CampusLife. Produk yang dikembangkan tersebut merupakan *mobile application* yang bertujuan untuk membantu civitas kampus mengakses informasi relevan tentang kampus mereka.

Saat ini CampusLife membutuhkan Web API yang digunakan untuk mendistribusikan data - data yang tersimpan di data store untuk selanjutnya diolah oleh mobile application. Aplikasi CampusLife akan didistribusikan ke pengguna yang menggunakan Android Smartphone. Smartphone merupakan salah satu *mobile computing devices* yang memiliki masa hidup baterai dan ketersediaan *bandwidth* yang terbatas. Dengan kedua keterbatasan tersebut proses pertukaran data antara Web API dan *mobile application* perlu dilakukan dengan efisien dalam hal ukuran data yang dikirimkan antara Web API dan *mobile application*.

Kata kunci: Web API, JSON, Format Serialisasi Data

Latar Belakang dan Masalah

CampusLife adalah *mobile information directory application* yang dikembangkan oleh LayangLayang Mobile¹ untuk menyediakan informasi yang relevan kepada civitas kampus. Setiap informasi yang ada di CampusLife akan didistribusikan melalui Web API² ke seluruh pengguna *smartphone* yang sudah memasang aplikasi CampusLife. Web API yang dibangun akan membuka akses ke *resource*³ yang disediakan oleh CampusLife, salah satunya Event⁴. *Resource* Event diolah berdasarkan data yang diambil dari *data store*⁵ yang tersimpan di salah satu layanan *Database as a Service*⁶ yang digunakan oleh LLM di AppFog⁷. *Resource* Event yang sudah diolah tersebut akan dikonsumsi oleh Web API *consumer* seperti perambah web atau *mobile application* dalam format serialisasi data tertentu.

Smartphone merupakan salah satu *mobile computing devices* yang memiliki masa hidup baterai dan ketersediaan *bandwidth* yang terbatas.[1] Dengan kedua keterbatasan tersebut proses pertukaran data antara Web API dan *mobile application* perlu dilakukan dengan efisien dalam hal ukuran data yang dikirimkan antara Web API dan *mobile application*.

Sampai awal tahun 2013 ini, sudah ada lebih dari 10 format serialisasi data⁸. Pemilihan format serialisasi data harus dilakukan dengan tepat agar ukuran data yang dikirimkan antara Web API dan *mobile application* bisa memiliki ukuran data yang kecil dan efisien. Dalam penelitian ini penulis akan memilih format serialisasi data JSON⁹ untuk digunakan dalam pengiriman data antara Web API dan *mobile application* CampusLife. Penulis memilih format serialisasi data tersebut karena tertarik untuk mengetahui langsung apakah JSON merupakan pilihan yang tepat. [3]

Dalam penelitian ini penulis akan mengkaji penerapan format serialisasi data JSON untuk digunakan dalam proses pertukaran data antara Web API dan CampusLife *mobile application*. Hasil akhir yang diharapkan adalah format serialisasi data JSON yang digunakan mampu memberikan ukuran *resource*¹⁰ yang kecil dan efisien dalam hal.[4]

¹ yang selanjutnya dalam proposal ini LayangLayang Mobile disebut sebagai LLM

² http://en.wikipedia.org/wiki/Web_API

³ http://en.wikipedia.org/wiki/Representational_state_transfer#Guiding_principles_of_the_interface

⁴ *Resource* yang berisi detail informasi Event yang disediakan di CampusLife

⁵ http://en.wikipedia.org/wiki/Data_store

⁶ http://en.wikipedia.org/wiki/Cloud_database

⁷ <http://www.appfog.com/>

⁸ http://en.wikipedia.org/wiki/Comparison_of_data_serialization_formats

⁹ <http://json.org/>

¹⁰ [http://en.wikipedia.org/wiki/Resource_\(Web\)](http://en.wikipedia.org/wiki/Resource_(Web))

Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dipaparkan, maka masalah yang akan diteliti dalam penelitian ini dapat dirumuskan sebagai berikut:

1. Bagaimana JSON dapat diterapkan secara efektif agar data yang dihasilkan dapat dikonsumsi oleh aplikasi *mobile* CampusLife dengan optimal?

Batasan Masalah

Untuk memperjelas ruang lingkup pelaksanaan penelitian, penulis memiliki batasan masalah meliputi:

- Pengembangan Web API hanya akan sampai pada tahap purwa-rupa.
- Pengembangan Web API hanya akan meliputi API untuk mengambil *Event resource*.
- JSON hanya mampu merepresentasikan data dalam bentuk teks, oleh karena itu data yang akan digunakan hanya terbatas pada data yang berbasis teks.
- Skema data *Event* akan disediakan oleh pihak LLM.
- Penulis akan melakukan demo untuk mengakses Web API melalui Android¹¹ *smartphone* yang sudah terpasang aplikasi *mobile* CampusLife. Aktivitas demo akan difokuskan pada pengambilan *Event resource* dari purwa-rupa Web API yang dibuat oleh penulis.
- Tidak membahas mengenai keamanan perangkat lunak, data dan jaringan.
- Pengembangan perangkat lunak menggunakan sebagian praktek dari Scrum dan tidak akan membahas Scrum secara komprehensif.

Tujuan

Diharapkan penerapan format serialisasi data JSON secara efektif dapat menghasilkan data yang optimal untuk dikonsumsi oleh aplikasi *mobile* CampusLife.

¹¹<http://www.android.com/>

Landasan Teori

Web API

Web API (*Application Programming Interface*) merupakan seperangkat *HTTP request message*¹² yang telah ditetapkan beserta definisi dari struktur *HTTP response message*¹³. Sementara Web API kadang - kadang dianggap sinonim untuk layanan web (*web service*), Web 2.0 aplikasi biasanya sudah pindah dari layanan web berbasis SOAP ke arah layanan web berbasis REST[5][9].

Layanan Web Berbasiskan REST

Representational State Transfer (REST) mendefinisikan seperangkat prinsip arsitektur dimana penulis dapat merancang layanan Web yang berfokus pada sistem *resource*, termasuk bagaimana keadaan *resource* dipanggil dan ditransfer melalui HTTP oleh berbagai macam klien yang ditulis dalam bahasa (pemrograman) yang berbeda. Jika diukur dengan jumlah layanan Web yang menggunakannya, REST telah muncul dalam beberapa tahun terakhir sebagai model layanan Web desain dominan. Bahkan, REST memiliki dampak besar di Web yang telah sebagian besar menggantikan desain antarmuka berbasis SOAP-WSDL dan karena REST memiliki gaya yang jauh lebih sederhana untuk digunakan[6].

JSON

JavaScript Object Notation (JSON) adalah format *data interchange* ringan. Sangat mudah bagi manusia untuk membaca dan menulis JSON. Sangat mudah untuk mesin (komputer) untuk mengurai dan menghasilkan JSON. JSON berbasiskan pada subset dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi 3 - Desember 1999. JSON merupakan format teks yang benar - benar bahasa independen tetapi menggunakan konvensi yang akrab bagi *programmer* dari keluarga bahasa C, termasuk C, C + +, C#, Java, JavaScript, Perl, Python, dan banyak lainnya. Properti ini membuat JSON sebagai bahasa *data interchange* yang ideal[7].

¹²<http://www.w3.org/Protocols/rfc2616/rfc2616-sec5.html>

¹³<http://www.w3.org/Protocols/rfc2616/rfc2616-sec6.html>

Prosedur Penelitian

Tahapan Penelitian

1. Identifikasi masalah
2. Perumusan hipotesis
3. Pengujian hipotesis
4. Kesimpulan

Lingkungan Pengembangan

Penelitian akan dibantu dengan menggunakan beberapa kakas dan teknologi berikut:

- NodeJS¹⁴, akan digunakan untuk membangun Web API.
- MongoDB¹⁵, akan digunakan sebagai basis data.
- Git¹⁶ dan Github¹⁷, akan digunakan untuk mengelola riwayat *source code* dan dokumen laporan serta dokumen teknis.
- AppFog¹⁸, akan digunakan sebagai infrastruktur untuk pengujian Web API.
- NetBeans IDE¹⁹, akan digunakan untuk penyuntingan *source code*.
- LaTeX²⁰, akan digunakan untuk penyuntingan dokumen laporan dan dokumen teknis.
- Laptop Asus Zenbook UX 32VD²¹ untuk pembangunan web API dan pembuatan dokumen laporan penelitian serta dokumen teknis.

¹⁴<http://nodejs.org/>

¹⁵<http://www.mongodb.org/>

¹⁶<http://git-scm.com/>

¹⁷<http://github.com/>

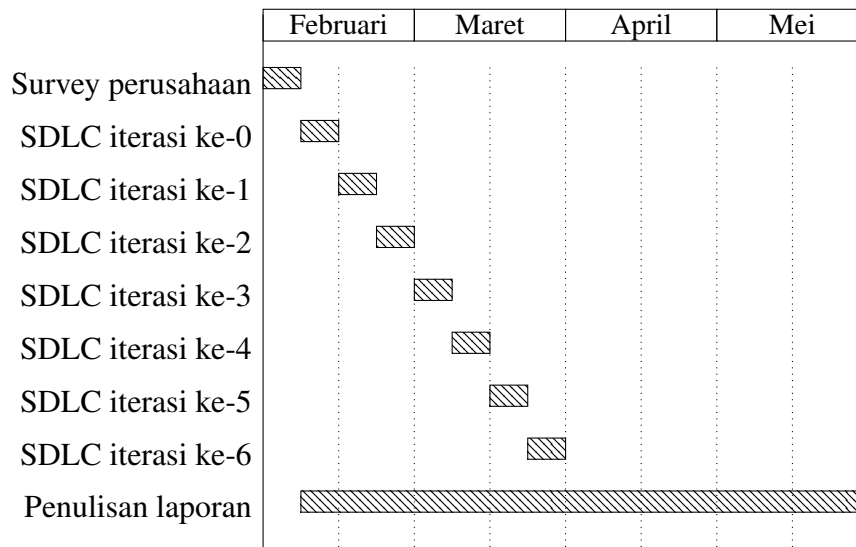
¹⁸<http://www.appfog.com/>

¹⁹<http://netbeans.org/>

²⁰<http://www.latex-project.org/>

²¹http://www.asus.com/Notebooks_Ultrabooks/ASUS_ZENBOOK_UX32VD/

Jadwal Kerja



SDLC (Software Development Life Cycle) terdiri dari tahap berikut:

1. Analisis Kebutuhan
2. Design
3. Implementasi
4. Pengujian

Penulis menggunakan metodologi pengembangan perangkat lunak *Agile*²² dengan metode yang dipilih adalah Scrum²³ dengan *timebox*²⁴ selama 1 minggu di setiap iterasi.

Usulan Pembimbing

Penulis mengharapkan proses penelitian ini dibimbing oleh ibu Sriyani Violina, S.T., M.T..

²²http://en.wikipedia.org/wiki/Agile_software_development

²³[http://en.wikipedia.org/wiki/Scrum_\(development\)](http://en.wikipedia.org/wiki/Scrum_(development))

²⁴<http://en.wikipedia.org/wiki/Timeboxing>

Bibliografi

- [1] Deepak, G., and Dr. Pradeep B S. "Challenging Issues and Limitations of Mobile Computing." *International Journal of Computer Technology and Applications* 3.1 (2012): Academic Journals Database. Web. 8 Jan. 2013.
- [2] *API Dashboard* <http://www.programmableweb.com/apis> diakses pada 27 Nopember 2012.
- [3] *Debate: JSON vs. XML as a data interchange format* <http://www.infoq.com/news/2006/12/json-vs-xml-debate> diakses pada 20 Januari 2012.
- [4] *JSON: The Fat-Free Alternative to XML* <http://www.json.org/xml.html> diakses pada 20 Januari 2012.
- [5] *Web API* http://en.wikipedia.org/wiki/Web_API diakses pada 20 Januari 2012.
- [6] *RESTful Web services: The basics* <http://www.ibm.com/developerworks/webservices/library/ws-restful/> diakses pada 14 September 2012.
- [7] *Introducing JSON* <http://www.json.org/> diakses pada 20 Januari 2012.
- [8] *Serialization* <http://en.wikipedia.org/wiki/Serialization>
- [9] *How REST replaced SOAP on the Web: What it means to you* <http://www.infoq.com/articles/rest-soap> diakses pada 14 September 2012.
- [10] *REST, And Now for Something Completely Different* <http://www.infoq.com/presentations/REST-And-Now-for-Something-Completely-Different> diakses pada 14 September 2012.