

# **Membangun Web API dengan menggunakan JSON sebagai format serialisasi data untuk menghasilkan ukuran data yang efisien**

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

23 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, XML

## Latar Belakang dan Masalah

CampusLife adalah *mobile information directory application* yang dikembangkan oleh LayangLayang Mobile<sup>1</sup> untuk menyediakan informasi yang relevan kepada civitas kampus. Setiap informasi yang ada di CampusLife akan didistribusikan melalui Web API<sup>2</sup> ke seluruh pengguna *smartphone* yang sudah memasang aplikasi CampusLife. Web API yang dibangun akan membuka akses ke *resource*<sup>3</sup> yang disediakan oleh CampusLife, salah satunya Event<sup>4</sup>. *Resource* Event diolah berdasarkan data yang diambil dari *data store*<sup>5</sup> yang tersimpan di salah satu layanan *Database as a Service*<sup>6</sup> yang digunakan oleh LLM di AppFog<sup>7</sup>. *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 data<sup>8</sup>. 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. Dalam penelitian ini penulis hanya akan mengambil 2 format serialisasi data untuk dibandingkan, yaitu JSON<sup>9</sup> dan XML<sup>10</sup>. Penulis memilih kedua format serialisasi data tersebut karena tertarik untuk mengetahui langsung apakah JSON merupakan alternatif yang tepat dibandingkan XML. Selain itu penulis tertarik untuk mempelajari lebih lanjut kedua format serialisasi data tersebut karena banyak perdebatan yang membandingkan antara JSON dan XML. [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*<sup>11</sup> yang lebih kecil

---

<sup>1</sup> yang selanjutnya dalam proposal ini LayangLayang Mobile disebut sebagai LLM

<sup>2</sup> [http://en.wikipedia.org/wiki/Web\\_API](http://en.wikipedia.org/wiki/Web_API)

<sup>3</sup> [http://en.wikipedia.org/wiki/Representational\\_state\\_transfer#Guiding\\_principles\\_of\\_the\\_interface](http://en.wikipedia.org/wiki/Representational_state_transfer#Guiding_principles_of_the_interface)

<sup>4</sup> *Resource* yang berisi detail informasi Event yang disediakan di CampusLife

<sup>5</sup> [http://en.wikipedia.org/wiki/Data\\_store](http://en.wikipedia.org/wiki/Data_store)

<sup>6</sup> [http://en.wikipedia.org/wiki/Cloud\\_database](http://en.wikipedia.org/wiki/Cloud_database)

<sup>7</sup> <http://www.appfog.com/>

<sup>8</sup> [http://en.wikipedia.org/wiki/Comparison\\_of\\_data\\_serialization\\_formats](http://en.wikipedia.org/wiki/Comparison_of_data_serialization_formats)

<sup>9</sup> <http://json.org/>

<sup>10</sup> <http://www.w3.org/TR/REC-xml/>

<sup>11</sup> [http://en.wikipedia.org/wiki/Resource\\_\(Web\)](http://en.wikipedia.org/wiki/Resource_(Web))

dibandingkan format serialisasi data XML.[4]

## Rumusan Masalah

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

1. Seberapa kecil ukuran data yang dapat dikirimkan apabila format serialisasi data yang digunakan adalah JSON?
2. Seberapa kecil ukuran data yang dapat dikirimkan apabila format serialisasi data yang digunakan adalah XML?

## Batasan Masalah

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

- Tidak membahas mengenai keamanan perangkat lunak, data dan jaringan.
- Pengembangan perangkat lunak menggunakan sebagian praktek dari Scrum dan tidak akan membahas Scrum secara komprehensif.
- Pengembangan Web API hanya akan tahap purwa-rupa.
- Pengembangan Web API hanya akan meliputi proses pengambilan dan penambahan *resource event*.
- Skema data *event* akan disediakan oleh pihak LLM.
- Penulis akan melakukan demo untuk mengakses Web API melalui Android<sup>12</sup> *smartphone* yang sudah terpasang aplikasi purwa-rupa dengan fungsi yang dibuat hanya membaca data dari Web API.

## Tujuan

Menerapkan format serialisasi data yang optimal pada pengiriman data antara Web API dan *mobile application*, sehingga dapat menghasilkan ukuran data yang efisien.

---

<sup>12</sup><http://www.android.com/>

## Landasan Teori

### Web API

Web API (*Application Programming Interface*) merupakan seperangkat *HTTP request message*<sup>13</sup> yang telah ditetapkan beserta definisi dari struktur *HTTP response message*<sup>14</sup>. 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 Berbasis 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].

### XML

*Extensible Markup Language* (XML) adalah bahasa markup yang mendefinisikan seperangkat aturan untuk pengkodean dokumen dalam format yang dapat dibaca

---

<sup>13</sup><http://www.w3.org/Protocols/rfc2616/rfc2616-sec5.html>

<sup>14</sup><http://www.w3.org/Protocols/rfc2616/rfc2616-sec6.html>

oleh manusia dan mesin. XML didefinisikan dalam *XML 1.0 Specification*<sup>15</sup> yang dibuat oleh W3C<sup>16</sup>[8].

Tujuan desain dari XML menekankan pada kesederhanaan, keumuman, dan kegunaan melalui Internet. Meskipun desain XML berfokus pada dokumen, namun XML secara luas digunakan untuk representasi dari struktur data yang berubah-ubah, misalnya dalam layanan web[8].

## Prosedur Penelitian

### Tahapan Penelitian

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

### Lingkungan Pengembangan

Penelitian akan dibantu dengan menggunakan beberapa kaskas dan teknologi berikut:

- NodeJS<sup>17</sup>, akan digunakan untuk membangun Web API.
- MongoDB<sup>18</sup>, akan digunakan sebagai basis data.
- Git<sup>19</sup> dan Github<sup>20</sup>, akan digunakan untuk mengelola riwayat *source code* dan dokumen laporan serta dokumen teknis.
- AppFog<sup>21</sup>, akan digunakan sebagai infrastruktur untuk pengujian Web API.
- NetBeans IDE<sup>22</sup>, akan digunakan untuk penyuntingan *source code*.
- LaTeX<sup>23</sup>, akan digunakan untuk penyuntingan dokumen laporan dan dokumen teknis.

---

<sup>15</sup><http://www.w3.org/TR/REC-xml/>

<sup>16</sup><http://www.w3.org/>

<sup>17</sup><http://nodejs.org/>

<sup>18</sup><http://www.mongodb.org/>

<sup>19</sup><http://git-scm.com/>

<sup>20</sup><http://github.com/>

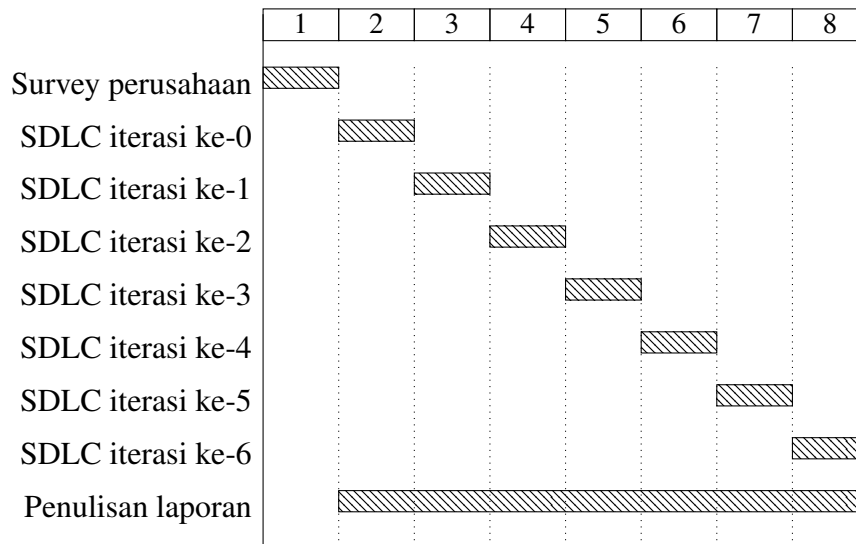
<sup>21</sup><http://www.appfog.com/>

<sup>22</sup><http://netbeans.org/>

<sup>23</sup><http://www.latex-project.org/>

- Laptop Asus Zenbook UX 32VD<sup>24</sup> untuk pembangunan web API dan pembuatan dokumen laporan penelitian serta dokumen teknis.

## 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*<sup>25</sup> dengan *timebox* selama 1 minggu di setiap iterasi.

## Usulan Pembimbing

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

<sup>24</sup>[http://www.asus.com/Notebooks\\_Ultrabooks/ASUS\\_ZENBOOK\\_UX32VD/](http://www.asus.com/Notebooks_Ultrabooks/ASUS_ZENBOOK_UX32VD/)

<sup>25</sup>[http://en.wikipedia.org/wiki/Agile\\_software\\_development](http://en.wikipedia.org/wiki/Agile_software_development)



# 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](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] *XML From Wikipedia* <http://en.wikipedia.org/wiki/XML> diakses pada 20 Januari 2012.
- [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.