

# PEMANFAATAN WEB SERVICE MOODLE BERBASIS REST-JSON UNTUK MEMBANGUN MOODLE ONLINE LEARNING EXTENSION BERBASIS ANDROID

**Bayu Wicaksono**  
Jurusan Informatika  
Universitas Sebelas Maret  
Jl. Ir. Sutami 36A Ketingan  
Surakarta 57126  
bayu.wicaksono777@gmail.com

**Wiharto, S.T., M.Kom.**  
Jurusan Informatika  
Universitas Sebelas Maret  
Jl. Ir. Sutami 36A Ketingan  
Surakarta 57126  
wi\_harto@yahoo.com

**Abdul Aziz, S.Kom., M.Cs.**  
Jurusan Informatika  
Universitas Sebelas Maret  
Jl. Ir. Sutami 36A Ketingan  
Surakarta 57126  
abdul\_7773@yahoo.com

## ABSTRAK

Moodle adalah salah satu *Learning Management System* yang telah memiliki fasilitas komunikasi sistem ke sistem menggunakan *web service*. Dalam sistemnya, Moodle menyediakan fitur “*Enable web service for mobile device*”, yang mana berfungsi untuk membuat servisnya dapat diakses menggunakan perangkat bergerak. Karena basis dari servis ini adalah sistem ke sistem, sehingga perlu diadakan penyesuaian fungsi *web service* yang akan digunakan untuk diimplementasikan pada perangkat bergerak (sistem ke *user*).

Penelitian ini membahas tentang bagaimana fungsi dari *web service* dari Moodle tersebut bisa dimanfaatkan secara lebih lanjut. Proses ini dimulai dari memperbaiki hak akses dari beberapa fungsi, kemudian pemilihan fungsi untuk akses secara *mobile*, hingga perancangan aplikasi yang akan mengakses fungsi dari *web service* tersebut. Dengan menggunakan *Engine REST* yang berformat data JSON akan di kembangkan sebuah prototipe aplikasi Android dengan melakukan *parsing* terhadap respons REST yang berupa JSON dan kemudian menampilkannya ke dalam sebuah tampilan antarmuka pada perangkat Android yang mudah digunakan, sehingga dapat membuat pengalaman menggunakan Moodle dengan cara yang berbeda.

Terdapat 23 servis berbasis REST-JSON, namun yang dapat digunakan untuk pengembangan ini hanya sembilan. Kesembilan fungsi tersebut didapat berdasarkan respons data yang mengandung informasi penting yang akan digunakan sebagai data untuk ditampilkan pada aplikasi *mobile*. Aplikasi *mobile* dibuat untuk menampilkan respons dari *web service* Moodle yang telah diolah melalui proses *parsing*. Hasil dari penelitian ini adalah sebuah prototipe aplikasi berbasis Android yang mengakses Moodle dengan menggunakan protokol *web service* REST-JSON. Dengan menggunakan respons sebagai parameter fungsi pada *web service*, pengalaman pengguna (*user experience*) akan sedikit berbeda dari pada mengakses Moodle menggunakan *web browser*. Akan tetapi, hal ini lebih memudahkan pengguna untuk mendapatkan informasi penting yang dibutuhkan.

**Kata Kunci:** Android, JSON, Moodle, REST, *Web Service*

## 1. PENDAHULUAN

Seiring dengan bertambahnya kebutuhan manusia terhadap akses internet, hampir semua lini kehidupan dan aktivitas manusia saat ini tak lepas dari teknologi komunikasi terutama internet. Termasuk dalam dunia pendidikan, yang telah mulai beradaptasi dengan perkembangan teknologi ini. Dari sinilah mulai terkenal istilah *e-learning*, yang secara umum berarti media pembelajaran secara elektronik.

Penggunaan *e-learning* secara luas telah menggunakan internet dalam proses belajar mengajar antara siswa dan guru. Dan salah satu *e-learning* berbasis internet yang paling banyak digunakan adalah Moodle. Yang mana Moodle merupakan LMS (*Learning Management System*), yang tidak hanya memfasilitasi proses pembelajaran, namun juga berbagai aspek yang berhubungan dengan pembelajaran.

Semenjak merilis Moodle versi 2.3, *web service* Moodle telah dapat mendukung 4 macam *engine web service*, diantaranya : AMF, REST, SOAP, dan XML-RPC. Di antara keempat *web service engine* tersebut, REST memiliki karakteristik yang paling mendukung dalam penggunaan *mobile* sebab membutuhkan *bandwidth* yang kecil dan *resources* yang sedikit [1]. Sedangkan REST, memiliki 2 format pengiriman data, yaitu dalam format XML dan format JSON. Di mana perbandingan performa dari keduanya, *web service* dengan basis JSON memiliki performa yang lebih baik [2].

Dalam perkembangan teknologi informasi modern yang bersifat “kapan saja dan di mana saja”, Moodle menyediakan fitur “*Enable web service for mobile devices*” untuk membuat servisnya dapat di akses menggunakan *mobile device*[3]. Sehingga dengan mengaktifkan fungsi ini, diharapkan pengguna perangkat bergerak, seperti *smartphone*, dapat dengan mudah mengakses Moodle.

Namun, pada dasarnya fitur “*Enable web service for mobile devices*”-ini merupakan pemanfaatan dari *web service basic* Moodle. Di mana *web service* tersebut dirancang untuk keperluan *System to System*, yaitu untuk interaksi sistem ke sistem. Maksudnya adalah sistem Moodle dengan sistem eksternal lain yang ingin diintegrasikan dengan sistem Moodle. Sehingga, *web service* ini pada dasarnya kurang cocok jika digunakan untuk keperluan penggunaan *mobile device*.

Maka dari itu, untuk tetap menggunakan servis ini pada *mobile device* perlu dilakukan analisis terhadap fungsi *web service* yang akan diaktifkan untuk keperluan penggunaan pada perangkat *mobile device* agar lebih optimal, baik dari sisi *client*, maupun dari sisi *server*. Dan untuk mendukung pemanfaatan tersebut, maka perlu dibuat sebuah model aplikasi untuk *smart client* untuk mengakses fungsi *web service* yang sesuai dengan kebutuhan penggunaan *mobile*.

*Mobile client* yang dimaksudkan di sini adalah *smartphone*. *Smartphone* dipilih sebab memiliki akses jaringan ke internet yang cukup baik dan memiliki kemampuan proses yang tinggi jika dibandingkan dengan *smart client non-smartphone*. Saat ini *smartphone* yang beredar di masyarakat Indonesia didominasi oleh *smartphone* dengan sistem operasi Android. Jumlah *smartphone* diperkirakan tumbuh dari 20% di tahun 2010 dan menjadi 40%

di tahun 2015, dan Android mengalami pertumbuhan paling pesat dibandingkan dengan *smartphone* dengan sistem operasi lainnya sebesar 43,3% [4].

## 2. LANDASAN TEORI

### 2.1 E-learning

Istilah *e-learning* mengandung pengertian yang sangat luas, beberapa diantaranya :

*E-learning* merupakan satu jenis belajar mengajar yang memungkinkan tersampainya bahan ajar ke siswa dengan menggunakan media internet, intranet atau media jaringan komputer lainnya [5]. Dalam definisi lain *e-learning* adalah sistem pendidikan yang menggunakan aplikasi elektronik untuk mendukung belajar mengajar dengan media internet, jaringan komputer, maupun komputer *standalone* [6]. *E-learning* adalah semua yang mencakup pemanfaatan komputer dalam menunjang peningkatan kualitas pembelajaran, termasuk di dalamnya penggunaan teknologi *mobile* seperti PDA (*Personal Digital Assistant*) dan MP3 *player*. Juga penggunaan *teaching materials* berbasis *web* dan *hypermedia*, multimedia CD-ROM atau *web sites*, forum diskusi, *e-mail*, blog, animasi pendidikan, simulasi, permainan, perangkat lunak manajemen pembelajaran dan lain-lain.[7].

Dari banyaknya definisi yang muncul, dapat kita simpulkan bahwa sistem atau konsep pendidikan yang memanfaatkan teknologi informasi dalam proses belajar mengajar dapat disebut sebagai *e-learning*. [8].

### 2.2 Moodle

Moodle merupakan singkatan dari *Modular Object Oriented Dynamic Learning Environment*, yang berarti tempat belajar dinamis dengan menggunakan model berorientasi objek[3]. Moodle adalah salah satu perangkat lunak manajemen pembelajaran yang cukup populer di masyarakat luas. Moodle merupakan *software open source* yang dirancang dengan menggunakan prinsip *pedagois*, yang membantu pendidik untuk menciptakan komunitas pembelajaran efektif. Prinsip – prinsip *pedagois* ini merupakan dasar dari pembangunan komunitas sosial yang merupakan platform khusus Moodle yang cocok untuk menciptakan komunitas pembelajaran.[6]

### 2.3 Web Service

W3C mendefinisikan *web service* sebagai sebuah sistem perangkat lunak yang dirancang untuk mendukung komunikasi dan interaksi antar mesin ke mesin (*Machine to Machine*) melalui sebuah *network* (jaringan). *Web service* juga termasuk *Web APIs* yang dapat diakses melalui jaringan seperti misalnya internet, dan dieksekusi melalui sebuah sistem jarak jauh sesuai dengan layanan yang diminta.

Definisi *web service* menurut W3C juga meliputi banyak sistem berbeda, tetapi pada umumnya lebih menyangkut pada *client* dan *server* yang berkomunikasi menggunakan XML yang memenuhi standar SOAP (*Simple Object Access Protocol*). Asumsi secara umum adalah pada terminologi terdapat deskripsi dari mesin yang layanannya disediakan oleh server, atau sama seperti konsep dari WSDL. WSDL bukan termasuk standard dari SOAP tetapi merupakan syarat mutlak untuk *client-side* otomatis pada *framework* Java dan .NET SOAP. Beberapa organisasi industri seperti WS-I

mengklaim baik SOAP dan WSDL sebagai definisi sari *web service*.

Selain SOAP dengan XML nya terdapat jenis engine *web service* lainnya yang banyak diimplementasikan pada aplikasi web, yaitu REST. REST *web service* atau yang kadang disebut RESTful *web service* atau RESTful API adalah *web service* yang mengimplementasikan arsitektur REST. Pada arsitektur REST, setiap service atau layanan dipandang sebagai sebuah *Resources* yang diidentifikasi melalui URL. *Web service* REST memiliki karakteristik sebagai berikut :

1. Menggunakan *method* HTTP secara eksplisit
2. Memiliki struktur direktori URI
3. Pesan yang ditransfer dalam format XML, JSON atau keduanya

### 2.4 REST-JSON

JSON (*Java Script Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari bahasa pemrograman *javascript*. JSON terdiri dari dua struktur, yaitu:

- Pasangan nama dengan nilai. Pada beberapa bahasa hal ini dinyatakan sebagai *object*, *record*, *struct*, *dictionary*, *hash table*, *keyed list* atau *associative array*.
- Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai *array*, *vector*, *list* atau *sequence*.

### 2.5 Android

Android adalah sebuah platform untuk perangkat *mobile* yang berbasis Linux mencakup sistem operasi, *middleware* dan aplikasi kunci. Pengembangan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Beberapa aplikasi inti Android antara lain *e-mail client*, SMS, kalender, peta, *browser*, kontak, dan lain-lain. Android adalah platform untuk perangkat *mobile* yang merupakan modifikasi *kernel* Linux versi 2.6, yang digunakan untuk layanan sistem inti seperti keamanan, manajemen memori, manajemen proses, *network stack*, dan model *driver*. Kernel juga bertindak sebagai lapisan abstraksi antara *hardware* dan seluruh *software stack*. Saat jurnal ini ditulis, Android telah mencapai versi 4.1 dengan *codename Ice Cream Sandwich*. Versi-versi sebelumnya antara lain versi 3.0 (Honeycomb), versi 2.3 (Gingerbread), versi 2.2 (Froyo), versi 2.0/2.1 (Eclair), versi 1.6 (Donut), versi 1.5 (Cupcake), dan versi 1.1.[9]

## 3. METODE PENELITIAN

### 3.1 Pengumpulan Data

Pada tahap ini akan di kumpulkan dokumentasi *function list* dari *web service* yang dapat di akses secara umum maupun *mobile*. Kemudian membuat data yang akan digunakan untuk simulasi terhadap masing-masing fungsi dari *web service* tersebut. Tabel 1 di bawah ini merupakan daftar fungsi hasil pengumpulan data yang kemudian akan digunakan untuk membangun aplikasi.

Tabel 1. Daftar fungsi *web service*

Nama Fungsi	Keterangan
core_course_create_courses()	Create new courses
core_course_get_courses()	Return course detail

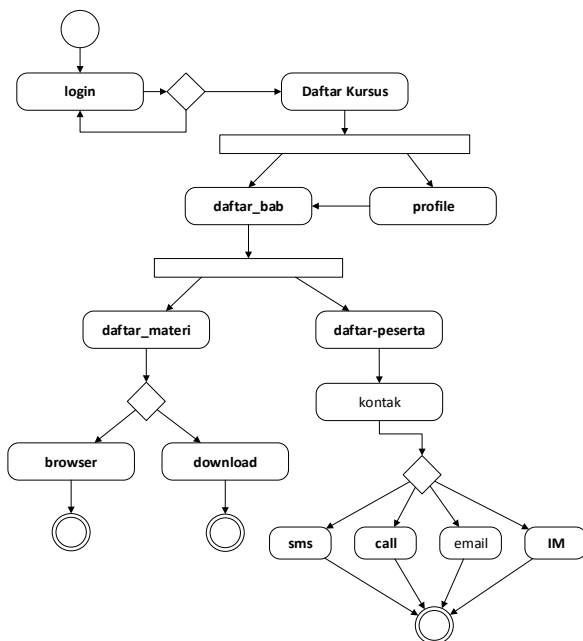
Tabel 1. Daftar fungsi *web service* (lanjutan)

Nama Fungsi Web Service	Keterangan
core_enrol_get_users_courses()	<i>get list of course ids that a user is enrolled in (if you are allowed to see that)</i>
enrol_manual_enrol_users()	<i>Manual enrol users</i>
core_files_get_files()	<i>browse moodle files</i>
core_files_upload()	<i>upload a file to moodle (DEPRECATED)</i>
core_group_add_group_members()	<i>Adds group Jember</i>
core_group_create_groups()	<i>Creates new groups</i>
core_group_delete_group_members()	<i>Deletes group members</i>
core_group_delete_groups()	<i>Deletes all specified groups</i>
core_group_get_course_groups()	<i>Returns all groups in specified course</i>
core_group_get_group_members()	<i>Returns group members</i>
core_group_get_groups()	<i>Returns group details.</i>

### 3.2 Analisis Sistem

Analisis dilakukan setelah mendapatkan respons dari masing-masing fungsi dengan menggunakan data yang telah dikumpulkan tersebut. Data dari respons *web service* tersebut kemudian digunakan sebagai data baru untuk merancang model aplikasi pada perangkat *mobile*.

Dengan memanfaatkan fungsi dari *web service* yang disediakan oleh Moodle, dirancang sebuah proses bisnis dan *Use-Case*. Gambar 1 di bawah ini adalah proses bisnis dari aplikasi ini.



Gambar 1. Proses Bisnis

Pada Gambar 1 tersebut di atas, dipetakan sebuah proses dalam pembuatan aplikasi *mobile* yang akan di gunakan untuk menampilkan data (respons) dari fungsi *web service*. Dengan menggunakan alur pemanfaatan respons untuk parameter berikutnya ditentukan fungsi apa saja yang dapat berjalan pada aplikasi ini, terdapat 2 jenis fungsi secara garis besar, yaitu fungsi yang menggunakan *web service* seperti terdapat pada Tabel 2, dan fungsi yang tidak menggunakan *web service* (Tabel 3).

Tabel 2. Daftar fungsi dengan *web service*

Fungsi Aplikasi	Keterangan
View Courses	Menampilkan daftar mata kuliah
View Chapters	Menampilkan daftar bab / pertemuan (per mata kuliah)
View Materials	Menampilkan daftar sub-bab / materi ajar
View Participant	Menampilkan daftar peserta dalam sebuah mata kuliah
View Profile	Melihat profil pengguna / user lain

Tabel 3. Daftar fungsi tanpa *web service*

Fungsi Aplikasi	Keterangan
Token	Melakukan autentikasi berbasis token
Setting Server	Melakukan konfigurasi server/ alamat website yang akan diakses
Communication	Melakukan kontak terhadap peserta lain dalam mata kuliah yang sama.

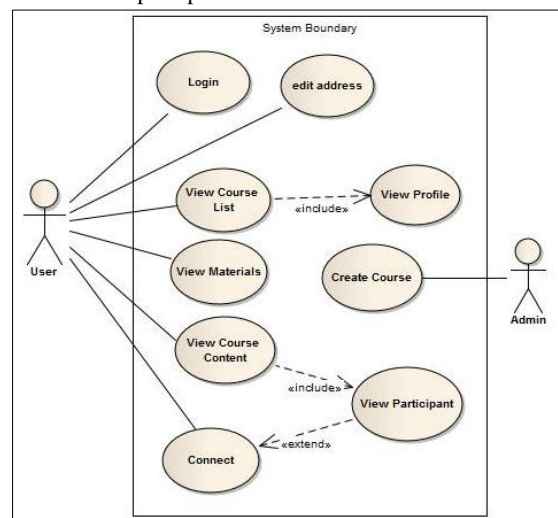
Untuk memudahkan pengembangan aplikasi dalam menerapkan *web service* ini, maka dibuatkan beberapa data set untuk mempermudah pembacaan respons dari *web service* yang akan digunakan. Tabel 4 berikut ini merupakan daftar *course* yang digunakan sebagai data set.

Tabel 4. Daftar course

No.	Nama	Kode	Format
1	Course One	C01	SCORM
2	Course Two	C02	Social
3	Course Three	C03	Topics
4	Course Four	C04	Weekly

### 3.3 Desain

Desain dari *prototype* aplikasi didasarkan pada *Use-Case* berikut seperti pada Gambar 2 di bawah ini.



Gambar 2. Use-Case Diagram

Desain ini menunjukkan bagaimana fungsi dari *web service* dimanfaatkan dalam aplikasi Android sebagai sebuah aplikasi *mobile*.

Terdapat beberapa fungsi yang tidak sepenuhnya memanfaatkan fungsi dari *web service*, yaitu: konfigurasi server, dan aktivitas kontak terhadap penggunaan lain serta fungsi login menggunakan verifikasi berbasis token.

## 4. HASIL DAN PEMBAHASAN

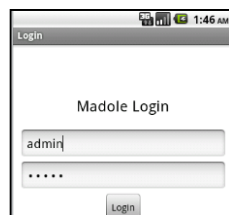
### 4.1 Gambaran Umum Aplikasi

Aplikasi ini adalah sebuah *prototype* aplikasi yang digunakan untuk mengakses *web service* dari situs pembelajaran *online* berbasis Moodle. Sebelum menggunakan aplikasi ini, pengguna diminta untuk memasukkan *username* dan *password* Moodle mereka, kemudian jika alamat *e-learning* belum tersimpan, maka akan diminta memasukkan alamat situs terlebih dahulu. Setelah masuk dalam aplikasi, maka *user* akan dapat melihat beberapa informasi yang diakses menggunakan *web service* Moodle.

### 4.2 Implementasi

Hasil dari penelitian ini adalah berupa sebuah *prototype* aplikasi *mobile* berbasis sistem operasi Android. Berdasarkan perancangan dari fungsi *web service* yang tersedia, telah dibuat sebuah *prototype*. Beberapa fungsional dari *web service* yang diimplementasikan adalah sebagai berikut:

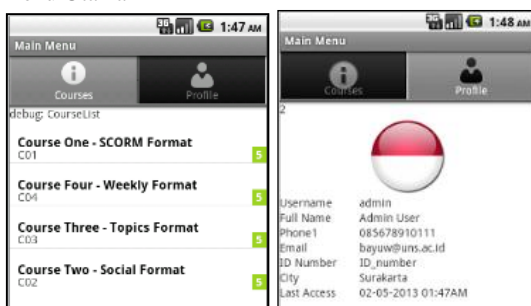
#### A. Login Screen



Gambar 3. Tampilan login

Gambar 3 di atas adalah halaman *login*, pada halaman ini digunakan fungsi *web service*, diantaranya, *create token* dan *web service* info. *Create token* merupakan fungsi dari *engine* Moodle yang ditransmisikan pada aplikasi *mobile* sebagai sebuah bentuk autentifikasi. Sedangkan *web service* info digunakan untuk mendapatkan *userid* dan informasi dari *website* Moodle yang di akses.

#### B. Menu Utama

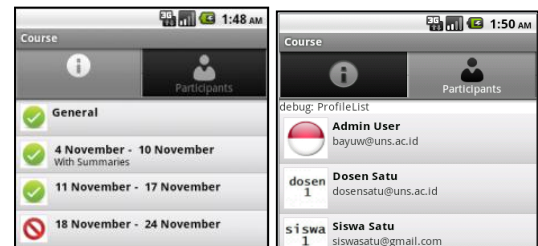


Gambar 4. Tampilan Kursus (kiri) dan Profil (kanan)

Pada Gambar 4 diatas terfapat dua halaman, halaman kursus, dan halaman profil. Pada halaman ini digunakan fungsi utama untuk mendapatkan daftar kursus/ mata pelajaran berdasarkan *user\_id* yang di dapat dari halaman *login*, dan dengan *user\_id* tersebut pada

tampilan ini disediakan tampilan profil dalam format tabulasi.

#### C. Daftar Sub Bab/Pertemuan



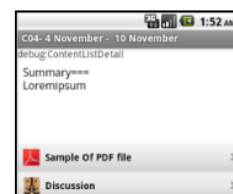
Gambar 5. Tampilan Sub Bab (kiri) dan Peserta (kanan)

Gambar 5 di atas merupakan tampilan daftar sub bab atau daftar pertemuan dari mata kuliah. Tampilan akan berbeda terhadap masing-masing jenis kursus. Terdapat beberapa perbedaan dari masing-masing *item*, yaitu antara bab yang memiliki *contetn*/materi dan bab yang memiliki rangkuman/ *summary*.

Pada tampilan ini juga terdapat tabulasi yang berisi daftar peserta kursus pada mata pelajaran tersebut, yang mana dapat diakses oleh pemilik akun yang telah melakukan *login*.

Pada daftar tersebut pengguna dapat melakukan beberapa opsi komunikasi, diantaranya melakukan panggilan langsung, mengirim pesan singkat (SMS), dan mengirim email. Sedangkan dengan menggunakan fungsi *web service*, dapat pula dilakukan opsi komunikasi berupa pengiriman instan *message*.

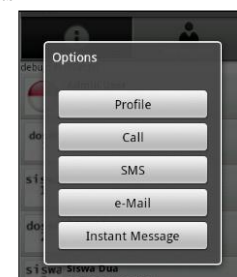
#### D. Daftar Material dan rangkuman materi



Gambar 6. Tampilan Materi dan Rangkuman

Gambar 6 di atas merupakan hasil pemetaan respons dari parameter *token*, *course\_id* dan *content\_id*. Secara alur, tampilan ini adalah lanjutan dari tampilan sub bab/pertemuan, di mana berisi daftar material pada sub bab/pertemuan yang bisa diunduh.

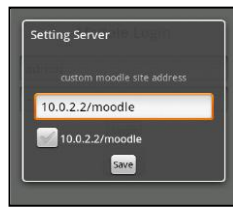
#### E. Opsi Komunikasi



Gambar 7. Tampilan

Pada Gambar 7 terdapat tampilan *pop-up screen*, di mana *user* bisa melakukan beberapa opsi komunikasi, diantaranya melakukan panggilan langsung, mengirim pesan singkat (SMS), dan mengirim email. Sedangkan dengan menggunakan fungsi *web service*, dapat pula dilakukan opsi komunikasi berupa pengiriman instan *message*.

#### F. Setting Server



Gambar 8. Tampilan setting server

Gambar 8 di atas merupakan tampilan *pop-up screen*, di mana *user* bisa melakukan konfigurasi terhadap server Moodle. Konfigurasi yang dimaksud adalah menentukan alamat *e-learning* Moodle yang digunakan. Sehingga dapat diakses menggunakan Android.

#### G. Tabel SQLite

Tabel 5. Tabel *user*

TABEL_USER	
PK	KEY_ID (inti)
	KEY_TOKEN(text)
	KEY_URL(text)
	KEY_CREATED_AT(datetime)

Dalam aplikasi yang diimplementasikan ini, dibutuhkan sebuah tabel dalam *database* yang reintegrasi dalam sistem operasi Android, aplikasi *database* tersebut adalah SQLite. Tabel 5 di atas ini digunakan untuk menyimpan rekaman data *login* pengguna aplikasi seperti toke dan alamat Url tempat *website* Moodle berada, sehingga dapat melakukan hubungan dengan berbagai situs Moodle yang telah mengaktifkan layanan *web service* miliknya.

### 4.3 Pengujian

Pada Moodle versi 2.3 terdapat 39 fungsi yang dapat diakses menggunakan layanan *web service*, namun setelah dilakukan pengujian, hanya ada 9 fungsi yang cocok untuk penggunaan Mobile.

Tabel 6 merupakan daftar fungsi yang dimiliki Moodle dari versi 2.0 yang memiliki respons yang digunakan untuk pengembangan aplikasi *mobile*.

Tabel 6. Daftar fungsi *web service*

versi	Nama Fungsi
2.0	core_course_get_courses()
2.0	core_enrol_get_users_courses()
2.0	core_files_get_files()
2.0	core_user_get_users_by_id()
2.1	core_enrol_get_enrolled_users()
2.1	core_message_send_instant_messages()
2.1	core_user_get_course_user_profiles()
2.1	core_webservice_get_site_info()
2.2	core_course_get_contents()

Tabel 7 berikut merupakan perbandingan waktu akses Clint terhadap *web service*, baik lokal maupun *hosted*.

Tabel 7. Daftar hasil pengujian *web service* (mili detik)

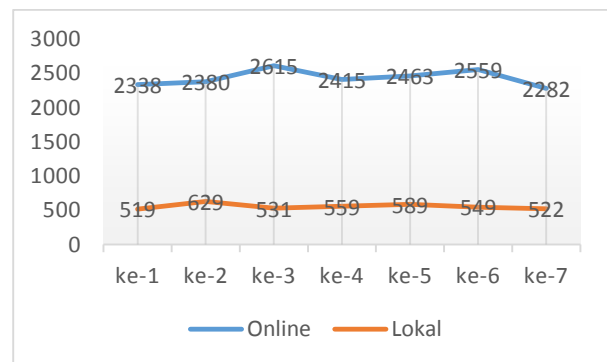
Nama fungsi	Online	Lokal
core_course_get_contents	2142	584
core_course_get_courses	3273	913

Tabel 7. Daftar hasil pengujian *web service* (mili detik)  
(lanjutan)

Nama fungsi	Online	Lokal
core_enrol_get_enrolled_users	3007	519
core_enrol_get_users_courses	2136	557
core_files_get_files	3750	554
core_message_send_instant_messages	2608	498
core_user_get_course_user_profiles	2505	468
core_user_get_users_by_id	1940	544
core_webservice_get_site_info	2581	683

Pada tabel tersebut terlihat perbedaan yang cukup signifikan lebih lama jika dilakukan pengujian pada aplikasi Online, hal ini tergantung pada kecepatan *provider* internet.

Dan berikut Gambar 9 di bawah ini adalah tampilan grafik secara rata-rata untuk semua fungsi yang dimiliki Moodle saat dilakukan 7 kali pengujian antara *Online* dan *Local*



Gambar 9. Grafik perbandingan (rata-rata)

## 5. KESIMPULAN

Aplikasi Moodle *Online learning extension* untuk pengguna Android dapat dikembangkan dengan memanfaatkan *web service* milik Moodle, namun tidak semua servis pada *web service* Moodle dapat digunakan, sebab beberapa fungsi memiliki *privilege* yang hanya boleh di akses oleh tingkat administratif. Sementara itu pemberian hak akses fungsi *web service* untuk *mobile* bersifat menyeluruh (sekali-gus), sehingga beberapa fungsi yang dapat merubah sistem tidak ditampilkan dalam aplikasi.

Terdapat 23 servis berbasis REST-JSON, namun yang dapat digunakan untuk pengembangan ini hanya sembilan. Ke-sembilan fungsi tersebut didapat berdasarkan respons data yang mengandung informasi penting yang akan digunakan sebagai data untuk ditampilkan pada aplikasi Mobile. Aplikasi *mobile* dibuat untuk menampilkan respons dari *web service* Moodle yang telah diolah melalui proses *parsing*.

Berdasarkan penelitian ini dapat disimpulkan bahwa: *web service* Moodle yang pada dasarnya dibangun untuk kebutuhan *system to system* dapat pula digunakan untuk *system to user*, namun mempunyai kelemahan yang terletak pada pembatasan hak akses saat akan menggunakan fungsi *web service* Moodle.



## 6. SARAN

Saran untuk pengembangan terkait berikutnya adalah: Perlu dikembangkan *plugin* baru yang berisi *web service* baru atau tambahan servis yang telah dikonfigurasi, sehingga dapat memaksimalkan fungsi Moodle pada aplikasi *mobile*, agar pengguna dapat menggunakan Moodle pada *mobile device* sebagaimana menggunakan Moodle pada aplikasi web, serta dilengkapi dengan fungsi integrasi ke berbagai sosial media untuk mempermudah komunikasi.

## 7. DAFTAR PUSTAKA

- [1] Castillo, P. A, "SOAP vs REST : Comparing a master-slave GA implementation", Neural and Evolutionary Computing Cornell University Library ,[on-line]: <http://arxiv.org/abs/1105.4978>, 2011.
- [2] Nurseotiv, N., Paulson, M., Reynolds, R., & Izurieta, C, "Comparison of JSON and XML Data Interchange Formats: A Case Study", ICSA 22nd International Conference on Computer Application in Industry and Engineering, 2009.
- [3] Moodle.org, "MOODLE.ORG", [on-line] <http://moodle.org/>, 2012.
- [4] Agung, W, [slide]: "Collaborative Innovation trough Telkom AppStore", Seminar Gunadharma, Jakarta, 2011.
- [5] Darin E, Hartly, "Selling e-learning American Society for Training And Development", New York, 2001.
- [6] LearnFrame.com, "Glossary of e-Learning Terms", [on-line]: <http://learnframe.com>, 2001.
- [7] Gilbert, S. M. & Jones, M. G. 2001. E-learning is enormous. Electric Perspectives, 26(3), 66-82.
- [8] Satria Wahono, Romi, "Pengantar e-Learning dan Perkembangannya", [ilmukomputer.org](http://ilmukomputer.org), artikel [on-line]: [www.ilmukomputer.org](http://www.ilmukomputer.org), 2008.
- [9] Safaat, Nazruddin, "Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android", Penerbit Informatika ,Bandung, 2010.