

PENGEMBANGAN WEBSERVICE SEBAGAI MEDIA PERTUKARAN DATA APLIKASI SCRUM PROYEK MANAJEMEN DENGAN TEKNOLOGI RESTFUL

¹Pratama Setya Aji(12018161),²Herman Yuliansyah(0512078304)

^{1,2}Program Studi Teknik Informatika Universitas Ahmad Dahlan

Jl. Prof. Dr. Soepomo, Janturan, Yogyakarta 55164

¹E-mail : pratamasetya99@gmail.com

²E-mail : herman.yuliansyah@tif.uad.ac.id

ABSTRAK

Scrum adalah sebuah kerangka kerja untuk mengembangkan dan mengelola produk kompleks, di mana pada saat bersamaan menghasilkan produk dengan nilai setinggi mungkin secara kreatif dan produktif. Metode komunikasi yang di gunakan pun ada dua, yaitu pertemuan secara langsung dan melalui media chatting aplikasi scrum, Namun keberadaan dua alternatif tersebut belum mampu memenuhi kebutuhan komunikasi dalam scrum, Untuk itu perlu dikembangkan sistem komunikasi yang berguna sebagai sarana monitoring proyek. Sarana itu dapat diwujudkan dalam bentuk komunikasi aplikasi mobile dan aplikasi web. Untuk mengkomunikasikan kedua hal tersebut dibutuhkan suatu webservice. Salah satu dari teknologi pengembangan webservice adalah restful.

Dari Permasalahan tersebut muncul gagasan untuk mengembangkan web service sebagai media pertukaran data aplikasi scrum proyek manajemen dengan teknologi restful. Aplikasi harus terkoneksi dengan basis data aplikasi web scrum agar dapat terintegrasi dengan aplikasi mobile scrum. Untuk itu sistem yang dibangun membutuhkan API KEY dan validasi data yang menjadi sarana pengamanan antara aplikasi dengan basis data. Pengujian aplikasi menggunakan metode white-box test dan stress test.

*Dari skripsi ini dihasilkan **Web Service Sebagai Media Pertukaran Data Aplikasi Scrum Proyek Manajemen Dengan Teknologi Restful** yang memiliki fitur keamanan akses melalui validasi data dan perlindungan url dengan api key sebagai alat bantu untuk menghubungkan aplikasi web dan aplikasi mobile. Serta aplikasi webservice yang dihasilkan telah teruji dapat diakses 1540 user dan berfungsi sebagaimana mestinya untuk media pertukaran data.*

Kata Kunci : Scrum, Manajemen, Webservice, Restful, Apikey.

A. PENDAHULUAN

Scrum dapat digunakan sebagai kerangka kerja untuk mengembangkan dan mengelola produk kompleks, di mana pada saat bersamaan menghasilkan produk dengan nilai setinggi mungkin secara kreatif dan produktif (Ken Schwaber, 2013). Kerangka kerja Scrum terdiri dari Team Scrum, serta peran-peran di dalamnya, acara-acara, artefak-artefak, dan aturan-aturan (Ken Schwaber, 2013). Setiap komponen di dalam kerangka kerja memiliki maksud tertentu dan peran penting demi keberhasilan penggunaan Scrum.

Masalah yang paling umum dari *scrum* diketahui bahwa tidak diterapkannya aturan-aturan *scrum* secara benar, seperti lebih memilih bekerja sekeras tenaga menyesuaikan *scrum* ke apapun yang sudah ada di organisasi daripada menyesuaikan kultur kerja saat ini ke kerangka kerja *scrum*. Penggunaannya di organisasi tersebut pun hanya akan bersifat sementara saja, tergantung berapa lama orang yang membawa *scrum* ke organisasi tersebut bertahan di dalam organisasi (Joshua Partogi, 2014).

Berdasarkan kutipan di atas, *frameworkscrum* lebih banyak mengedepankan komunikasi antar *stake holder* yang terlibat, seperti dalam kegiatan *meeting*, *review*, dan *Sprint Retrospective*. Untuk itu perlu dikembangkan sistem komunikasi yang berguna sebagai sarana *monitoring* proyek. Sarana itu dapat diwujudkan dalam bentuk aplikasi *mobile* dan aplikasi *web*. Melalui teknologi *web* dan *mobile* tim proyek dapat melakukan *meeting*, *review*, dan *Sprint Retrospective* kapan saja dan dimana saja. Untuk menghubungkan aplikasi *web* dan aplikasi *mobile* itu kita perlu yang namanya *webservice*. *Web service* dirasa mampu mengatasi permasalahan tersebut. *Web service* diartikan sebagai sebuah antar muka (*interface*) yang menggambarkan sekumpulan operasi-operasi yang dapat diakses melalui jaringan, misalnya *internet* dalam bentuk pesan XML (Kreger, 2001). Sekalipun mirip dengan proses *Application Programming Interface* (API) berbasis *web*, *web services* memiliki keunggulan karena dapat dipanggil dari jarak jauh melalui *internet*, pemanggilan *web services* bisa menggunakan bahasa pemrograman apa saja, dan dalam *platform* apa saja, sementara API hanya bisa digunakan untuk *platform* tertentu (Lucky, 2008).

Salah satu teknologi *web service* saat ini dikenal dengan *rest*. *REST* yang disebut juga dengan *RESTful* pada dasarnya merupakan setiap *URL* unik dan berupa representasi dari beberapa objek yang digunakan untuk mendapatkan konten-konten objek dengan menggunakan *HTTP Request*. Keuntungan utama *web service REST* yaitu *lightweight*, tidak membutuhkan *XML markup* tambahan, hasilnya dapat dibaca dengan mudah oleh manusia (*human readable result*), mudah untuk dikembangkan, tidak membutuhkan toolkit. Keuntungan lain dari antarmuka *RESTful* salah satunya *request* dan respon dapat dipendekkan. Namun berdasarkan hasil uji coba diperoleh *Restful* memiliki kelemahan dari sisi keamanan, karena pada prinsipnya *request* ke suatu *RESTful web service* sebenarnya adalah suatu *HTTP Request*, dan *REST* hanya memiliki 1 *URL* untuk setiap *service*, sehingga ketika *URL* dapat diketahui maka semua data yang ada di dalamnya dapat dimanipulasi dengan semua metod yang ada pada *REST*. Selain permasalahan tersebut juga ditemukan masalah lain, yaitu validasi data yang dikirim oleh aplikasi *mobile* perlu di validasi sehingga data yang tersimpan di *database* merupakan data yang sesuai dengan keperluannya.

Berdasarkan latar belakang tersebut maka dibuatlah penelitian Pengembangan *Web Service* sebagai media pertukaran data aplikasi *Scrum* Proyek Manajemen dengan teknologi *restful*. *Sistem* ini merupakan aplikasi yang menjadi penghubung komunikasi antara aplikasi berbasis *web* dengan aplikasi *mobile scrum*. Dengan adanya aplikasi ini diharapkan mampu membantu *user* untuk menggunakan *tool scrum* sebaik mungkin secara aman.

B. KAJIAN PUSTAKA

Pada bagian ini akan dipaparkan mengenai kajian terdahulu yang disajikan sebagai bahan acuan dasar dalam pembuatan penelitian. Selain kajian terdahulu, akan dijelaskan tentang kajian teori yang mendukung penelitian ini. berikut beberapa kajian terdahulu yang diacu :

Penelitian terdahulu yang dilakukan Martinus Raditia Sigit Surendra (2014), tentang implementasi *PHP web service* sebagai penyedia data aplikasi *mobile*. *Webservice* ini dibangun untuk menyediakan data aplikasi *mobile*, yang dapat mempermudah aplikasi *mobile* dalam komunikasi data dengan *databaseserver*. *Webservice* ini di bangun dengan teknologi *SOAP* yang belum di lengkapi fitur keamanan aksesnya. Penelitian ini menghasilkan *webservice* yang dapat menyediakan data untuk perangkat aplikasi *mobile*.

Penelitian terdahulu yang dilakukan Hartati Deviana (2011) tentang Penerapan *XMLWeb Service* Pada Sistem Distribusi Barang. Penelitian ini bertujuan untuk menerapkan *web service* pada sistem pengelolaan distribusi barang. Penelitian ini lalu menghasilkan sistem informasi yang mampu mengintegrasikan aplikasi dan *platform* dari seluruh cabang sistem yang ada.

Penelitian terdahulu yang dilakukan Yuli Purwati dan Fandy Setyo H (2014), penelitian ini dilakukan untuk merancang sistem berorientasi objek aplikasi *webservice* untuk *interoperabilitas* data pada sistem informasi akademik dengan sistem informasi kepegawaian. Penelitian ini bertujuan untuk membuat *protoype* layanan validasi antara dua sistem di atas dengan teknologi XML.

C. METODE PENELITIAN

1. Objek penelitian

Obyek penelitian yang akan dibahas penelitian ini adalah aplikasi *web* untuk manajemen proyek berbasis *scrum* dan aplikasi *mobile daily scrum*. Aplikasi tersebut membutuhkan media pertukaran data antara *web* manajemen *scrum* dan *mobiledaily scrum* yang dilengkapi fitur keamanan dalam pengaksesanya.

2. Metode Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh data atau dokumen yang dibutuhkan dalam penelitian. Data yang diperoleh akan diproses sesuai dengan kebutuhan penelitian. Adapun metode yang dilakukan untuk mengumpulkan data dalam penelitian ini yaitu Studi Pustaka dan Observasi. Metode studi pustaka dilakukan dengan mencari dan mempelajari pustaka berupa buku, makalah, jurnal, artikel, termasuk pula pustaka-pustaka *digital* mengenai pengembangan *web service* yang aman. Metode oservasi dilakukan dengan mengembangkan dan menguji *webservice* yang sesuai dengan kebutuhan di aplikasi *scrum*.

3. Metode Pengembangan

Metode yang digunakan untuk merancang bangun aplikasi web untuk Manajemen Proyek berbasis Scrum ini ialah menggunakan kerangka kerja Scrum. Scrum merupakan salah satu kerangka kerja dari metode pengembangan perangkat lunak Agile.

Adapun tim Scrum yang terlibat adalah sebagai berikut :

- a. Product Owner : Herman Yuliansyah,S.T.,M.Eng.
- b. Scrum Master : Adi Alamsyah

- c. Developer Team :
 - 1) Sarah Nurul Qudsiah
 - 2) Muhammad Reza
 - 3) Pratama Setya Aji
 - 4) Adi alamsyah

4. Jalannya Penelitian

Adapun jalannya penelitian yang akan dilakukan pada skripsi ini adalah sebagai berikut :

a. Penentuan Product Backlog

Product backlog ditentukan oleh Product Owner. Product Owner menentukan dengan mencatat semua kebutuhan yang dibutuhkan oleh aplikasi. Daftar Product Backlog langsung diurutkan sesuai prioritasnya. Product Backlog tidak dibatasi dan akan menjadi acuan atau sumber utama dari semua kebutuhan yang harus ada pada aplikasi ini. Product Backlog didokumentasikan menggunakan Trello. Product Backlog akan digunakan pada saat Sprint Planning Meeting untuk menentukan tugas-tugas yang akan dikerjakan sebagai Sprint Backlog.

b. Penentuan Sprint

Sprint ditentukan dengan mengacu pada Product Backlog yang telah ditentukan sebelumnya. Developer Team melaksanakan Sprint Planning Meeting untuk menentukan Sprint yang akan dikerjakan oleh Developer Team. Dalam Sprint Planning Meeting juga menentukan estimasi pengerjaan setiap Sprint oleh Developer Team dalam satuan jam. Sprint didokumentasikan di Trello. Sprint Planning Meeting akan menghasilkan Sprint yang siap untuk dikerjakan.

c. Pengerjaan Sprint dan Daily Scrum

Setelah Sprint Planning Meeting selesai dan sudah menghasilkan Sprint, selanjutnya Developer Team akan segera bekerja menyelesaikan Sprint yang telah diperoleh dari Sprint Planning Meeting. Setiap Sprint dikerjakan oleh satu orang anggota Developer Team. Pekerjaan yang dilakukan meliputi pembuatan user interface dan pembuatan kode program.

Daily Scrum dilakukan setiap hari dan di jam yang sama. Daily Scrum dilaksanakan di satu tempat atau dengan menggunakan media messagging Whatsapp. Waktu pelaksanaan Daily Scrum maksimal 15 menit dan hadirin yang terlibat dalam Daily Scrum adalah Scrum Master dan Developer Team. Daily Scrum membahas apa yang telah dilakukan Developer Team selama 24 jam sebelumnya, menentukan apa yang akan dikerjakan Developer Team untuk 24 jam selanjutnya dan menyampaikan hambatan yang di dapat selama menyelesaikan Sprint Backlog atau hambatan yang akan menghalangi Developer Team untuk mencapai Sprint Goal.

d. Sprint Review

Sprint Review dilaksanakan diakhir Sprint untuk meninjau pekerjaan yang telah dikerjakan oleh Developer Team. Sprint Review dihadiri oleh tim Scrum dan Stakeholder dan waktu pelaksanaannya maksimal empat jam. Tim Scrum dan Stakeholder berkolaborasi untuk membahas Sprint yang telah dikerjakan dan diselesaikan oleh Developer Team. Developer Team

mendemonstrasikan Sprint yang sudah mereka selesaikan dan menjawab pertanyaan-pertanyaan dari apa yang telah Developer Team demonstrasikan. Selain itu, tim Scrum dan Stakeholder juga berkolaborasi untuk membahas pekerjaan selanjutnya sebagai masukan untuk Sprint Planning berikutnya.

e. Sprint Retrospective

Sprint Retrospective dilakukan untuk memberi kesempatan Tim Scrum meninjau dirinya sendiri dan membuat perencanaan untuk pekerjaan selanjutnya. Dilaksanakan dengan waktu maksimal tiga jam dan dilaksanakan sesudah Sprint Review dan sebelum Sprint Planning Meeting berikutnya. Tim Scrum mengidentifikasi dan mengurutkan hal-hal yang utama yang berjalan baik dan hal-hal yang berpotensi meningkatkan serta membuat rencana implementasi dengan tujuan peningkatan cara-cara kerja Tim Scrum.

5. Pengujian Sistem

Software Testing yang dilakukan pada aplikasi yang akan dibuat pada penelitian ini ialah :

a. White-box Test

White-box test adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian. Adapun metode yang akan digunakan dalam white-box test yaitu menggunakan unit testing yang disediakan oleh framework PHP Codeigniter.

b. Stress-Test

Pengujian ini bertujuan untuk melihat apakah perangkat lunak secara keseluruhan mampu menangani kebutuhan sumberdaya yang tidak normal (mencakup kuantitas, frekuensi, maupun volume).

D. HASIL DAN PEMBAHASAN

1. Jalannya Penelitian

Adapun jalannya penelitian yang telah dilakukan pada skripsi ini adalah sebagai berikut :

a. Hasil *ProductBacklog*

Product Backlog aplikasi web untuk manajemen proyek berbasis *Scrum* yang ditentukan oleh *Product Owner* berasal dari *User Stories*. *User Stories* menggambarkan deskripsi sederhana dan singkat fitur aplikasi dari pengguna atau *customer* yang menginginkan fitur baru dari perangkat lunak. Berikut adalah *User Stories* dan *ProductBacklog* yang ditentukan secara urut menurut prioritasnya :

Tabel 4.1. Tabel *User Stories* dan *Product Backlog*

Prioritas	User Stories	Product Backlog
1	Sebagai pengguna aplikasi <i>mobile</i> , saya dapat melakukan <i>login</i> dengan mengetikkan <i>username</i> berupa email dan <i>password</i> minimal	<i>Log in mobile</i>

	6 karakter.	
2	Sebagai pengguna, saya dapat mengatur ulang <i>password</i> dengan mengisikan alamat email di aplikasi web dan aplikasi <i>mobile</i> dan akan mendapat notifikasi via email yang berisi <i>password</i> sementara.	<i>Reset password mobile</i>
3	Sebagai pengguna, saya dapat menggunakan mengedit profil di aplikasi <i>mobile</i> dan aplikasi web.	<i>Profile mobile</i>
4	Sebagai pengguna di aplikasi <i>mobile</i> , saya dapat melihat dashboard project yang sedang dikerjakan.	<i>Dashboard project mobile</i>
5	Sebagai pengguna, saya menerima notifikasi berupa email atas undangan ke dalam scrum project dan notifikasi di dashboard aplikasi web dan aplikasi mobile.	Notifikasi <i>invite tim scrum mobile</i>
6	Sebagai pengguna, saya dapat melihat history daily scrum	daily scrum mobile
7	Sebagai pengguna, saya dapat mengirim dan menerima pesan di web dan mobile	chat mobile
8	Sebagai pengguna, saya dapat melihat tim dari project scrum di web dan mobile	tim scrum mobile
9	Sebagai pengguna, saya dapat melihat sprint yang ada pada web dan mobile	Sprint mobile
10	Sebagai pengguna, saya dapat melakukan sinkronisasi data yang ada pada mobile dan web secara aman	Authentifikasi webservice

Hasil *productbacklog* yang telah ditentukan oleh *Product Owner* pada tabel 4.1. berjumlah 10 *product backlog*. Adapun *product backlog* pada prioritas 1 sampai dengan nomor 10 dikerjakan oleh penulis. Setiap *product backlog* pada setiap *developer* memiliki prioritas masing-masing dan dikerjakan runtut sesuai urutan prioritas.

b. Hasil Sprint

Penentuan *sprint* ditentukan saat mengadakan *Sprint Planning Meeting* yang diadakan di Ruang Lab. Jaringan Kampus 3 UAD jalan Janturan. Dilaksanakan pada tanggal 4 Mei 2016 pukul 10.00 WIB dan diadakan selama 3 jam. Adapun *sprint* yang dihasilkan saat *SprintPlanningMeeting* tersebut adalah sebagai berikut :

Tabel 4.2. Tabel *Product Backlog* dan *Sprint Backlog*

No	Product Backlog	Sprint Backlog	Estimasi
1	<i>Log in mobile</i>	<i>service Log in</i>	30 jam
2	<i>Reset password mobile</i>	<i>service Reset password</i>	24 jam
3	<i>Profile mobile</i>	<i>service tampil profile</i>	24 jam

		<i>service edit profile</i>	32 jam
4	<i>Dashboard Project mobile</i>	<i>service tampil dashboard</i>	32 jam
5	Notifikasi <i>invite tim mobile</i>	<i>service tampil notifikasi</i>	32 jam
6	<i>Daily scrum mobile</i>	<i>service tampil history daily</i>	24 jam
7	<i>Chat mobile</i>	<i>service kirim chat</i>	32 jam
		<i>service tampil chat daily</i>	24 jam
		<i>service tampil chat by date</i>	24 jam
8	Tim scrum mobile	<i>service tampil tim scrum</i>	32 jam
9	<i>Sprint mobile</i>	<i>service tampil sprint todo</i>	24 jam
		<i>service tampil sprint doing</i>	24 jam
		<i>service tampil sprint done</i>	24 jam
		<i>service tampil sprint verified</i>	24 jam
10	<i>Authentifikasi webservice</i>	Configurasi <i>basic auth HTTP</i>	12 jam
		Configurasi <i>api-key</i>	12 jam
Total			430 jam

Hasil sprint yang telah ditentukan pada acara sprint planning meeting yang ditunjukkan pada tabel 4.2. telah dirinci dengan estimasi yang ditentukan pada tiap-tiap sprint. Estimasi setiap sprint ditentukan sendiri oleh developer masing-masing sesuai dengan kemampuannya. Adapun *sprint* pada prioritas 1 sampai dengan nomor 10 dikerjakan oleh penulis dengan total estimasi yaitu 430 jam. *Sprint* dikerjakan oleh masing-masing *developer* sesuai dengan urutan prioritas.

c. Software Testing

Pada tahap ini akan dilakukan pengujian terhadap webservice sebagai media pertukaran data aplikasi scrum proyek manajemen dengan teknologi restful, apakah sudah memenuhi persyaratan sebagai suatu aplikasi web. Software Testing yang dilakukan pada aplikasi yang akan dibuat pada penelitian ini ialah :

1) White-box Test

White-box Test menggunakan pengujian unit. Pengujian ini menggunakan *tools Unit testing codeigniter* yang dikembangkan oleh *codeigniter* untuk otomatisasi pengujian. Terdapat 19 fungsi di dalam aplikasi ini, tetapi pengujian hanya dilakukan terhadap 5 fungsi yang ada pada aplikasi *webservicescrum* proyek manajemen. fungsi tersebut adalah fungsi *servicelogin*, *service tampil profile*, *service tampil chat*, *service tampil dashboard*, dan *service sprint*.

2) Stress Test

Pada tahap ini dilakukan *stress test* dengan *tool Jmeter* terhadap aplikasi *web service* untuk manajemen proyek Scrum dengan mengukur tingkat kinerja dalam menangani banyak *request* pada satu waktu secara bersamaan. Tes tingkat kinerja ini dijalankan menggunakan *environment test* sebagai berikut :

1. **Server Webservice :**

a. RAM 512 MB

- b. 1 Core Processor
 - c. 20 GB SSD
 - d. 1 TB Transer
 - e. OS Ubuntu 16.04
 - f. MariaDB
 - g. Teknologi Restful Webservice
2. **Client Webservice :**
- a. RAM 4 GB
 - b. Core i3 Processor
 - c. 120 GB SSD
 - d. OSX 10.11.5
 - e. Koneksi 10 MB/S

Tabel 4.12. Hasil *StressTest*

Users	Loop count	Period (s)	CPU		Median (ms)	Min (ms)	Max (ms)	Error (%)
			Usage Server (%)	Average (ms)				
10	1	1	4,3	143	139	133	166	0
50	1	1	12,3	254	258	154	363	0
100	1	1	52,2	799	811	277	1294	0
500	1	1	67,4	4422	4183	354	8792	0
1000	1	1	89,2	8831	8676	364	126215	0,70
2000	1	1	100,0	16262	16124	387	168293	23

Dari 6 *samplestress test*, dapat dilihat bahwa tingkat kinerja maksimal dari aplikasi *webservice* aplikasi *scrum* dengan parameter *environment* yang sudah ditentukan sebelumnya dapat menangani 2000 *request* dalam waktu satu detik secara bersamaan dengan tingkat *error* 23% yang artinya 1540 *request* dapat terlayani, akan tetapi setelah itu *server* mengalami *crash* dan mati. Sehingga dapat disimpulkan bahwa tingkat kinerja aplikasi *webservice* untuk Manajemen Proyek berbasis *Scrum* maksimal diakses oleh 1540 *user* dalam satu waktu.

E. PENUTUP

1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

- a. Telah dihasilkan aplikasi *webservice* yang sesuai kebutuhan aplikasi *mobile* scrum dan *web* scrum proyek manajemen untuk pertukaran data berdasarkan hasil 100%*passed white box test*.
 - b. Aplikasi *webservice* sudah melewati *stresstest* dan dapat direquest sebanyak 1540 *user* per detik serta memiliki fitur perlindungan terhadap proses pengaksesanya melalui *apikey*, dan *basicauth* sebagai media validasi datanya.
2. Saran
- Saran yang dapat disampaikan dari hasil peneitian ini adalah sebagai berikut :
1. Dalam media validasi dan autentikasi melalui *basic auth* masih dapat di tingkatkan ke metode auth yang lebih *secure* seperti *digest*, *Oauth*, atau *webservice signature*.
 2. Untuk *request* yang menggunakan *method get* sebaiknya parameter yang tampil di url di enkripsi terlebih dahulu.

F. DAFTAR PUSTAKA

- Holmes, Steve J. and Walsh, Robert T. (2005). Conducting Effective Proyek Manajemen Maturity Assessment Interviews. IMSI TECH 2005. Integrated Manajemen Systems, Inc., Ann Arbor, MI. Diunduh dari www.imsi-pm.com/home/library/conducting_assessment.pdf, 13 November 2008.
- Gray, Clifford F. and Larson, Eric W. (2006). Proyek Manajemen: The Managerial Process 3th Edition, Singapore: The McGraw-Hill Companies. Inc.
- Kerzner, Harold. (2001 a). Proyek Manajemen: A Systems Approach to Planning, Scheduling, and Controlling - 7th Editions. Canada: John Wiley & Sons.
- Microsoft Corp. (2000) *Application Service Provider: Evolution and Resources*, White Paper, USA.
- Bougettaya, A., Sheng, Q.Z., *Advanced Web Services*, Springer 2014
- Richardson, L., Ruby, S., *RESTful Web Services*, O'Reilly , California 2007
- Siswoutomo, W., 2004, *Membangun web service Open Source menggunakan PHP*, PT Elex Media Komputindo, Jakarta
- Kreger, H., 2001, *Web-services Conceptual Architecture (WSCA 1.0)*, IBM Software Group, USA
- Sandoval, Jose., 2009, *RESTful Java Web Service* , First Edition, Packt Publishing, 2009.