

ANALISIS PERBANDINGAN METODE SOAP DAN REST YANG DIGUNAKAN PADA FRAMEWORK FLASK UNTUK MEMBANGUN WEB SERVICE

¹M Gilvy Langgawan Putra, ²M Ihsan Alfani Putera

¹ Program Studi Informatika, Jurusan Matematika dan Teknologi Informasi
Institut Teknologi Kalimantan

² Program Studi Sistem Informasi, Jurusan Matematika dan Teknologi Informasi, Institut
Teknologi Kalimantan

Email : ¹gilvy.langgawan@itk.ac.id, ²ihsanalfani@itk.ac.id

Abstrak. SOAP singkatan dari Simple Object Access Protocol. SOAP adalah protokol untuk pertukaran informasi dengan desentralisasi dan terdistribusi. SOAP dibangun dengan menggunakan protokol komunikasi HTTP. Karena HTTP didukung oleh semua browser dan server, maka SOAP dapat berkomunikasi dengan berbagai aplikasi meskipun terdapat perbedaan sistem operasi, teknologi, dan bahasa pemrogramannya. Peran SOAP di dalam teknologi web service adalah sebagai protokol pemaketan untuk pesan-pesan (messages) yang digunakan secara bersama oleh aplikasi-aplikasi penggunaannya. Flask adalah microframework. Bila dibandingkan dengan Django, Flask jauh lebih ringan dan cepat karena Flask dibuat dengan ide menyederhanakan inti framework-nya seminimal mungkin. Dengan tagline “web development, one drop at a time”, Flask dapat membantu kita membuat situs dengan sangat cepat meskipun dengan librari yang sederhana. Pada Flask, kita tidak akan menemukan konsep-konsep besar seperti MVC. Namun Flask masih menyediakan fungsi-fungsi dasar seperti routing. REST masih cukup baru, sedangkan SOAP telah merevolusi RPC dan lebih terbuka dibanding batasan-batasan yang ada di versi sebelumnya. REST lebih kepada filosofi lama, ketimbang sebuah teknologi yang baru. Tetapi dalam kenyataannya datang kemudian dalam teknologi

Kata Kunci: Flask , SOAP, REST.

Sistem terdistribusi merupakan kumpulan dari beberapa komputer otonom yang mana terhubung oleh jaringan dengan sebuah software, software tersebut dirancang untuk menghasilkan beberapa fasilitas komputerisasi yang saling terintegrasi yang mana dianggap sebagai komputer tunggal.

Sistem terdistribusi memiliki karakteristik penting dalam pendistribusiannya, yang mana terdiri dari jenis komputer yang berbeda dan cara komunikasi yang berbeda juga, tanpa diketahui user.

Adapun karakteristik lainnya adalah sistem terdistribusi, user dan aplikasi dapat saling berinteraksi secara konsisten serta dengan tampilan yang sama, kapanpun dan dimanapun si user mengakses. Aplikasi sistem terdistribusi dapat berjalan bersamaan pada prosesor yang berbeda. Dewasa ini perkembangan sistem berbasis jaringan semakin banyak dikembangkan. Salah satu perkembangannya adalah teknologi web

services. Web services menggunakan standar yang tidak terikat pada platform (platform-neutral) dan tidak terikat pada bahasa pemrograman yang digunakan (language-neutral). Dengan demikian, web services memudahkan beberapa aplikasi atau komponennya untuk saling berhubungan dengan aplikasi lain dalam sebuah organisasi maupun di luar organisasi, web service dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan dimana sebuah database ditanamkan, dibuat dalam bahasa apa sebuah aplikasi yang mengkonsumsi data, dan di platform apa sebuah data itu dikonsumsi. web service mampu menunjang interoperabilitas. Sehingga web service mampu menjadi sebuah jembatan penghubung antara berbagai sistem yang ada. Yang mana nantinya dengan analisis ini mahasiswa akan mengetahui bagaimana api seperti REST dan SOAP berjalan di framework Flask.

Studi Literatur

Web Service

Pengertian sederhana web service adalah aplikasi yang dibuat 2 agar dapat dipanggil atau diakses oleh aplikasi lain melalui internet atau intranet dengan menggunakan XML sebagai format pengiriman pesan.[1]

Orang berpendapat web service semacam web site, tetapi itu bukan demikian. Ada perbedaan antara web service dengan website.

Website :

- Memiliki web interface
- Dibuat untuk berinteraksi langsung dengan user

Web Service :

- Tidak memiliki tampilan atau interface yang bagus
- Dibuat untuk berinteraksi langsung dengan aplikasi yang lain baik beda sistem operasi atau konsep sekalipun.

Web service menyimpan dan melakukan pertukaran datanya dalam format XML, karenanya menjadi multi platform dalam hal accessibilitynya. Karena XML merupakan suatu format dokumen yang berbasis teks, maka web service memungkinkan berlangsungnya komunikasi antar aplikasi yang berbeda dengan platform yang berbeda pula dan dapat menghemat waktu dalam komunikasi antara aplikasi dengan service penyedia. Beberapa vendor luar negeri mulai berkolaborasi satu sama lain dengan konsep web services, diantaranya : IBM, Microsoft, SUN, ORACLE. Salah satu contoh web services yang sudah jadi dan dipakai adalah Web Services keluaran Microsoft (Microsoft Passport) web services untuk user name dan password yang sudah dipasang di web site Microsoft dan HOTMAIL.

Flask adalah microframework yang dipelopori oleh Armin Ronacher. Flask jauh lebih ringan dan cepat karena Flask dibuat dengan ide menyederhanakan inti framework-nya seminimal mungkin. Dengan tagline “web development, one drop at a time”, Flask dapat membantu kita membuat situs dengan sangat cepat meskipun dengan librari yang sederhana.

SOAP

(Simple Object Access Protocol) adalah sebuah XML-based mark-up language untuk pergantian pesan diantara aplikasi. SOAP berguna seperti sebuah amplop yang digunakan untuk pertukaran data object didalam network. SOAP mendefinisikan empat aspek didalam komunikasi: Message envelope, Encoding, RPC call convention, dan bagaimana menyatukan sebuah message didalam protokol transport [2].

Sebuah SOAP message terdiri dari SOAP Envelop dan bisa terdiri dari attachments atau tidak memiliki attachment. SOAP envelop tersusun dari SOAP header dan SOAP body, sedangkan SOAP attachment membolehkan non-XML data untuk dimasukkan kedalam SOAP message, di-encoded, dan diletakkan kedalam SOAP message dengan menggunakan MIME-multipart. HTTP berbasis API berarti API yang diekspos sebagai salah satu atau lebih HTTP URI dan respon berupa XML/JSON. Skema respon dapat dikustomasi untuk setiap objek.

REST

REST (REpresentational State Transfer) merupakan sebuah teknik di arsitektur software untuk sistem terdistribusi seperti World Web Wide . REST tidak memerlukan parsing XML dan tidak memerlukan sebuah header pesan ke dan dari penyedia layanan. Hal ini pada akhirnya menggunakan mengurangi penggunaan bandwidth [3].

RESTful web service atau juga dikenal dengan nama RESTful Web API merupakan sebuah web service yang di implementasikan dengan menggunakan http dengan menggunakan prinsip-prinsip REST. Service yang digunakan menggunakan method milik http antara lain GET, PUT, POST or DELETE.

API

API (application programming interface) adalah sekumpulan perintah, fungsi, komponen, dan protokol yang disediakan oleh sistem operasi ataupun bahasa pemrograman tertentu yang dapat digunakan oleh programmer saat membangun perangkat lunak

XML

XML (Extensible Markup Language) adalah bahasa markup untuk keperluan umum yang disarankan oleh W3C untuk membuat dokumen markup keperluan pertukaran data antar sistem yang beraneka ragam. XML merupakan kelanjutan dari HTML (HyperText Markup Language) yang merupakan bahasa standar untuk melacak Internet [4]

XML didesain untuk mampu menyimpan data secara ringkas dan mudah diatur. Kata kunci utama XML adalah data (jamak dari datum) yang jika diolah bisa memberikan informasi.

XML menyediakan suatu cara terstandarisasi namun bisa dimodifikasi untuk menggambarkan isi dari dokumen. Dengan sendirinya, XML dapat digunakan untuk menggambarkan sembarang view database, tetapi dengan suatu cara yang standar.

JSON

JSON adalah sebuah format universal yang digunakan pada web service untuk meningkatkan kolaborasi dan komunikasi antar sistem. Komunikasi antar sistem tersebut menggunakan sebuah format yang bersifat universal, sehingga walaupun sistem yang berhubungan adalah berbeda platform, sistem operasi maupun berbeda bahasa pemrograman akan tetap bisa saling komunikasi.

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999.

WSDL

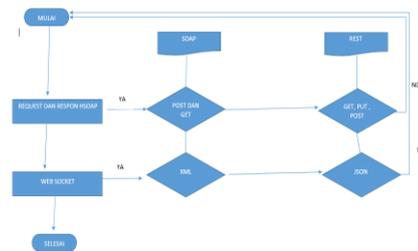
WSDL merupakan kependekan dari Web Services Description Language. WSDL adalah bahasa berbasis XML untuk mendeskripsikan Web services dan bagaimana untuk mengaksesnya. WSDL membantu konsumen web service untuk memakai/menggunakan layanan web service. Tanpa WSDL web service akan terkunci dan tidak berguna. WSDL menspesifikasikan

lokasi service dan operasi (methods) yang disediakan oleh web service.

Sebagai protokol komunikasi dan format pesan yang distandarkan pada komunitas web, WSDL menjadi lebih tepat dan penting untuk mendeskripsikan model komunikasi di beberapa langkah yang terstruktur (dalam memprogram web service). WSDL menyediakan kebutuhan ini dengan mendefinisikan bentuk XML untuk mendeskripsikan layanan jaringan sebagai kumpulan-kumpulan titik akhir (endpoints) komunikasi yang mampu melakukan pertukaran pesan.

I. Metodologi

Rancangan Pengujian Sistem



Gambar 1 Flowchart Rancangan Pengujian Sistem

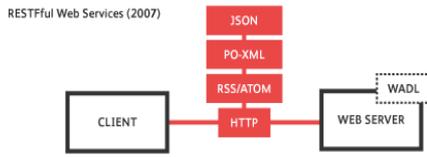
Adapun pada perancangan pengujian nantinya akan ada 2 model pengujian yang menjadi parameter analisis SOAP dan RESTful yang berjalan di framework Flask ini, yaitu terdiri dari :

- Request & respons REST dan SOAP

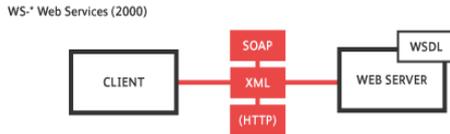
Dalam pengujian analisis perbandingan request dan respon Yang mana akan ada request dari klien dan ada reaksi atau respon dari server yang diakses oleh klien. Untuk REST sendiri akan diuji request dan responnya terdiri dari GET, PUT, dan POST sedangkan di SOAP akan di uji POST dan GET.

- Web socket rest & soap

Dalam pengujian analisis perbandingan rest dan soap dalam hal web socket, yang mana ada perbedaan web socket antara rest dan soap saat ada interaksi antara klien dan web server. Pada Gambar 2 yang menunjukkan interaksi antara client dan web server yang terjadi pada REST, dan Gambar 3 menunjukkan SOAP.



Gambar 3.2 REST

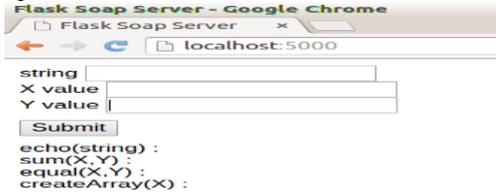


Gambar 3.3 SOAP

II. Hasil dan Pembahasan
Pengujian Request dan Respon REST dan SOAP

- SOAP

Untuk pengujian soap ini kami mengakses web sederhana yang tersedia, yang mana menunjang analisis kami yang yaitu terdiri dari POST dan GET untuk SOAP ini. dan berdiri di framework Flask. Tampilan web service FLASK yang menggunakan SOAP seperti dibawah ini:



Gambar 2 Index

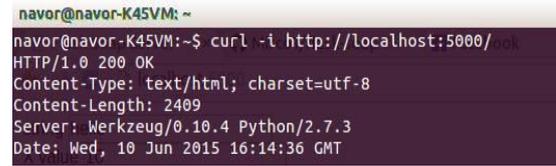
Gambar 2 tersebut bahwa saat mengakses halaman index.htm akan tampil seperti itu, itu merupakan salah satu dari request klien, dari request klien tersebut mana akan tampil respon dari server yang akan menampilkan seperti dibawah ini.



Gambar 3 Respon GET dari klien

Gambar 3 merupakan hasil dari request oleh klien yang ada pada gambar 4.1, sehingga respon dari server akan menampilkan seperti pada gambar 4.2, yang dimaksud adalah saat client ingin mengakses

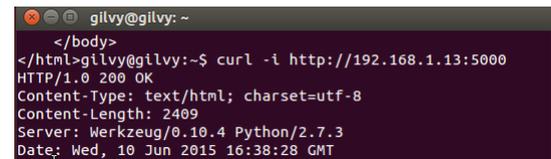
http://localhost:5000 maka klien akan mendapatkan apa yang diaksesnya.



Gambar 4 Detail dengan curl

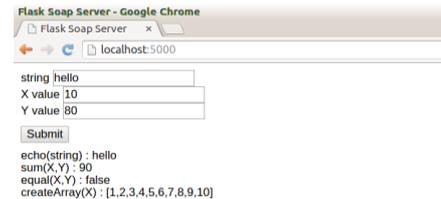
Gambar 4 itu menunjukkan saat client mengakses melalui terminal sama seperti gambar 4.2, namun disini disebutkan bahwa untuk mengakses web servicenya maka mendapatkan content-length sebesar 2409, akses dari http://localhost:5000.

Adapun akses klien dari komputer lain yaitu http://192.168.1.13:5000 menghasilkan content length yang sama, seperti gambar di bawah ini.



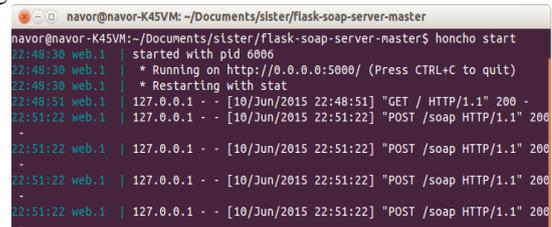
Gambar 5 akses dari klien lain

Untuk post pada SOAP yang mana dari web service tersebut menunjukkan pada gambar 5, bahwa pada field yang kosong diisi sesuai yang diinginkan fieldnya sendiri. Seperti gambar dibawah ini :



Gambar 5 Index 2

Gambar 5 menunjukkan bahwa klien ingin merequest hasil dari pengisian field tersebut yang mana nantinya yang diserver terjadi proses POST, yang ditunjukan pada gambar dibawah ini.

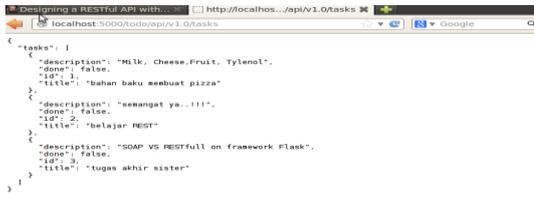


Gambar 6 Respon POST dari klien

Gambar 6 bahwa saat klien minta respon untuk menampilkan hasil maka ada proses yang terjadi didalamnya, yang mana server akan mempost notification POST.

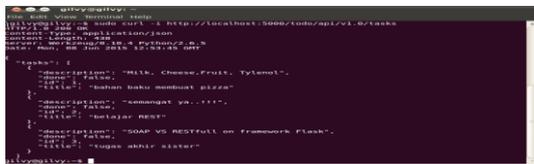
- REST

Untuk pengujian REST ini kami mengakses web sederhana yang tersedia, yang mana menunjang analisis kami yang yaitu terdiri dari POST, GET dan PUT untuk REST ini. dan berdiri di framework Flask. Tampilan web service FLASK yang menggunakan REST seperti dibawah ini:



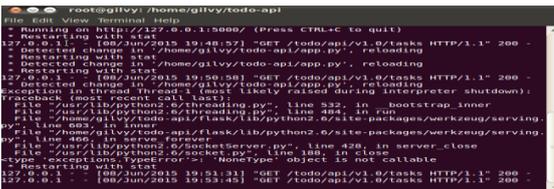
Gambar 7 REQUEST REST

Gambar 7 itu menunjukkan bahwa saat mengakses rest pada web browser.atau saat klien minta request ke server. Kita bisa melihat detailnya seperti pada gambar dibawah ini.



Gambar 8 REQUEST REST TERMINAL

Gambar 8 menunjukan bahwa saat klien meminta request ke server dengan menggunakan terminal maka diketahui konten length nya bernilai rendah, yaitu sebesar 438. Adapun respon dari servernya saat klien meminta request seperti gambar dibawah ini.



Gambar 9 RESPON GET REST SERVER

Gambar 9 menunjukan bahwa saat klient mengakses web, maka reaksu dari server akan mendapatkan GET, kenapa mendapatkan GET, karena klient mengakses

dengan localhost maupun secara ip address maka klient akan mendapatkan halaman browser, dan dari respon server akan menampilkan GET.

Untuk pengujian selanjutnya yaitu untuk request dan respon POST pada REST, yang mana request dan respon untk post itu sama halnya dengan GET, akan tetapi saat siklient meminta sebuah request dari server dari yang telah didapat atau GET tadi maka akan ada respon untuk POST, seperti ditunjukkan gambar dibawah ini, yang menjelaskan saat siklient meminta atau merequest halaman lain.



Gambar 10 REQUEST POST REST KLIEN

Gambar 10 tersebut menunjukkan bahwa saat dia mengakses content dengan tertentu, atau si klient meminta untuk mengakses suatu halaman maka akan tampilan baru sesuai dari request klien tersebut, dan respon dari servernya akan seperti gambar di bawah ini, karena dari klien yang meminta akses suatu halaman maka server akan merespon dengan POST tersebut. Akan di tunjukan pada gambar dibawah ini.



Gambar 11 RESPON POST REST SERVER

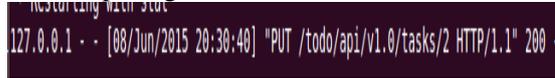
Gambar 11 Untuk selanjutnya request dan respon untuk PUT pada rest sama seperti sebelum-sebelumnya bahwa untuk ini hanya megunakan method PUT, mari kita langsung saja saat klien meminta atau request dengan PUT akan ditunjukkan pada penjelasan gambar dibawah ini, dengan klien meminta perintah PUT, pada local host dengan tambahan put, seperti dibawah ini.



Gambar 12 REQUEST PUT REST KLIEN

Gambar 12 menunjukan bahwa dengan method PUT dengan mengakses ID =

2 maka akan muncul seperti pada gambar tersebut yang menampilkan atau mengambil halaman dengan id yang bernilai 2. Dan respon dari server nya akan menampilkan PUT, seperti gambar dibawah ini.



Gambar 13 RESPON PUT REST SERVER

Gambar 13 menunjukkan bahwa saat klien mengakses dengan id=2, yang berarti klien ingin mengambil data dari id yang bernilai sama dengan 2, maka respon dari servernya akan seperti pada Gambar 13.

- Hasil

Tabel 1 RESPON and REQUEST

SN	SOAP	REST
	REQUEST AND RESPON	REQUEST AND RESPON
1	2409	121
2	2409	146
3	2409	233
4	2409	118
5	2409	120
6	2409	26
7	2409	438



Gambar 14 Grafik Request and Respon SOAP and REST

Dari data yang diperoleh yang ada pada Tabel 1 dan hasil grafik pada Gambar 14 bahwa untuk web service yang berbasis Flask dengan menggunakan SOAP dan REST api dari grafik tersebut, REST memiliki performance yang lebih bagus dibandingkan dengan SOAP untuk pengamatan request dan respon untuk web service.

a. Web socket rest & soap

Untuk pengujian web socket disini, yaitu bagaimana saat klien merequest ke server seperti POST, GET dan PUT,

penggunaan web socketnya bagaimana terhadap request klient tersebut. Ada pun hasil pengujian dari web socket dari REST dan SOAP sebagai berikut:

- REST

Socket untuk rest api pada penggunaan metode GET yang mana dia berada pada gambar dibawah ini merupakan salah satu web socket yang digunakan untuk REST api.

```

@app.route('/todo/api/v1.0/tasks/<int:task_id>', methods=
['GET'])
def get_task(task_id):
    task = [task for task in tasks if task['id'] == task_id]
    if len(task) == 0:
        abort(404)
    return jsonify({'task': task[0]})
    
```

Gambar 15 Method REST

Dari Gambar 15 itu merupakan untuk metode get, yang apabila klien untuk mendapatkan akses ke get dia, akan di route kan dengan memasukan ID setelah http://localhost:5000/todo/api/v1.0/tasks/2 yang mana angka 2 itu merupakan id yang ingin diakses oleh klien. Bahwa dengan mengakses id=2 maka akan tampil di browser dengan data yang memiliki id = 2.

- SOAP

Socket untuk SOAP disini, dia menggunakan XML, yang mana sebagai penjabar bahasanya menggunakan WSDL, dalam mengakses sebuah web service yang menggunakan api SOAP ini, dengan menggunakan XML, tanpa ada lemparan yang page baru, pada web yang kami uji ini. saat kami untuk mengakses webnya atau get dan post ke http://localhost:5000/ langsung ke halaman tersebut, dan mengakses untuk perhitungan, tetap seperti itu juga, tidak seperti menggunakan REST.



Gambar 16 XML pada SOAP.

III. Simpulan

1. Dari data yang diperoleh yang ada pada Tabel 1 dan hasil grafik pada gambar 14 bahwa untuk web service yang berbasis Flask dengan menggunakan SOAP dan REST api dari grafik tersebut, REST memiliki performance yang lebih bagus dibandingkan dengan SOAP untuk pengatasan request dan respon untuk web service.
2. Dengan menggunakan socket json dan xml lebih baik menggunakan socket XML yang mana pade untuk mendapatkan beberapa metode tidak diketahui oleh client yang mengakses web browser. Tanpa harus ada lemparan halaman lainnya.

IV. Daftar Pustaka

- [1]. M.Muchlen , J.V.Nickerson and K.D.Swenson ,” Developing web services choreography standards—the case of REST vs. SOAP,”

- Decision Support System vol 40 ,p.9-29,2005.
- [2]. Q.Sheng , X.Qiao , A.Valiakos , C.Szabo , S.Bourne and X.Xu ,” Web services composition: A decade’s overview,” Information Sciences vol 280 ,p.218-238,2014.
- [3]. C.Pautasso ,” RESTful Web service composition with BPEL for REST,” Data & Knowledge Engineering vol 68 ,p.851-866,2009.
- [4]. A.Muralidhar,V.Pattabiraman,” An Efficient Association Rule Based Clustering of XML Documents”Procedia Computer Science vol 50,p.401-407.