

## **PENERAPAN WEB SERVICES DAN REGULAR EXPRESSION UNTUK VERIFIKASI ALAMAT BERBASIS HASIL PENELUSURAN SITUS KODEPOS PT POS INDONESIA**

**Tegar M. Aji<sup>1</sup>, Djalal Er Riyanto<sup>2</sup>, Helmie Arif Wibawa<sup>2</sup>**  
Jurusan Ilmu Komputer/Informatika FSM Universitas Diponegoro

### **ABSTRAK**

Situs Kodepos PT Pos Indonesia (KPPI) merupakan situs yang menyediakan fasilitas penelusuran data alamat dan kodepos berdasarkan input lokasi dan kota. Data kodepos dan kelengkapan alamat lainnya pada situs KPPI ini diperlukan oleh berbagai pihak. Namun demikian, situs KPPI belum menerapkan antarmuka yang memungkinkan datanya dapat diakses oleh situs lain. Untuk mengatasi masalah tersebut diperlukan adanya suatu aplikasi pihak ketiga yang dapat melakukan fungsi penelusuran data ke situs KPPI dan menyediakan antarmuka berupa Web Services (suatu metode pengiriman pesan/data melalui dokumen terstandardisasi).

Perangkat lunak Verifikasi Alamat (VA) dikembangkan dalam rangka menyediakan antarmuka pengaksesan data situs KPPI. VA dikembangkan menggunakan bahasa scripting PHP versi 5.3.5. Cakupan fungsional VA yaitu menerima input dari situs klien, melakukan penelusuran ke situs KPPI, menguraikan halaman penelusurannya dengan Regular Expressions (suatu metode untuk memanipulasi data teks), serta mengembalikan hasil data alamat kepada situs klien tersebut. Dengan adanya VA, maka proses pengaksesan data situs KPPI oleh situs klien dapat berlangsung.

**Kata kunci:** verifikasi alamat, web services, regular expressions, situs KPPI

### **ABSTRACT**

Situs Kodepos PT Pos Indonesia (KPPI) is a website which provides addresses and postcodes data retrieval based on location and city inputs. Postcodes data and another address complement data on KPPI website are necessary for some stakeholders. However, the KPPI website did not implement an interface which allows their data accessible by other sites. To overcome these problems, the existence of a third-party application which able to do data retrieval from the KPPI website and to provide Web Services (a method of sending messages/data through standardized documents) interfaces was required.

Verifikasi Alamat (VA) software was developed in order to provide data interface access to the KPPI site. VA was developed using the PHP scripting language version 5.3.5. The scope of VA's functionals are receives inputs from client websites, perform searches to KPPI website, parsing its search result page using Regular Expressions (a method to manipulate text data), and to return the results to the concerned client website. In the presence of VA, the data access of KPPI website by client websites were able to took place.

**Keywords:** address verification, web services, regular expressions, KPPI website

1. *Mahasiswa Jurusan Ilmu Komputer/Informatika FSM Universitas Diponegoro*
2. *Dosen Jurusan Ilmu Komputer/Informatika FSM Universitas Diponegoro*

## 1. Pendahuluan

### 1.1. Latar Belakang

Dewasa ini, bisnis jual-beli secara *online* semakin tumbuh pesat. Melalui situs jual-beli *online*, pembeli dan penjual yang terpisah oleh jarak yang jauh dapat dengan mudah bertransaksi melalui *E-commerce*. *E-commerce* merupakan sistem yang memungkinkan perdagangan dijalankan secara *online*. Pada proses bisnis *online* terdapat beberapa komponen penting. Salah satu komponen penting penunjang bisnis *online* adalah jasa pengiriman barang. Terdapat banyak penyedia jasa pengiriman baik milik pemerintah maupun swasta. Pemanfaatan jasa pengiriman akan lebih optimal apabila pengguna jasa selalu menyertakan alamat tujuan dengan lengkap dan benar. Apabila alamat tidak disertakan dengan lengkap dan benar, akan berakibat barang kiriman terkirim dengan jangka waktu yang lebih lama atau bahkan tidak dapat sampai tujuan.

Dalam rangka mengantisipasi hal tersebut, PT Pos Indonesia telah menyediakan situs penelusuran data alamat dan kodepos. Meskipun demikian, situs tersebut saat ini hanya bisa dimanfaatkan dengan cara mengetikkan alamat ke dalamnya. Dengan semakin canggihnya teknologi situs *E-commerce*, hal ini menjadi tidak praktis karena diperlukan waktu tambahan untuk melakukan *input* alamat secara manual. Akan lebih praktis apabila situs *E-commerce* memiliki fungsi verifikasi alamat pengiriman yang terintegrasi dengan situs penelusuran data alamat dan kodepos PT Pos Indonesia.

Namun demikian, terdapat beberapa hal yang menyebabkan pengaksesan data alamat pada situs PT Pos Indonesia oleh situs *E-commerce* tidak dapat dilaksanakan. Salah satu penyebabnya adalah situs *E-commerce* memiliki *platform* atau bahasa pemrograman yang berbeda-beda. Selain itu, situs PT Pos Indonesia belum menyediakan antarmuka pemrograman yang memungkinkan pengaksesan data oleh situs lain.

Sementara itu, pada dunia teknologi informasi dikenal suatu fasilitas bernama *Web Services* yang memungkinkan pertukaran data antar aplikasi berbeda *platform* melalui antarmuka protokol yang terstandardisasi. Aplikasi-aplikasi yang akan bertukar data tersebut perlu menyediakan antarmuka terstandardisasi. Apabila antarmuka tersebut belum tersedia, maka diperlukan *tool* tambahan untuk menguraikan data. Salah satu metode untuk memanipulasi data berformat teks yaitu *Regular Expression (Regex)*. Perpaduan antara *Web Services* dengan *Regex* akan menghadirkan suatu aplikasi yang memungkinkan pertukaran data antar aplikasi yang berbeda *platform* dan belum memiliki antarmuka terstandardisasi.

## 1.2. Rumusan Masalah

Berdasarkan uraian pada bagian latar belakang, maka permasalahan pada tugas akhir ini yaitu bagaimana menerapkan teknologi *Web Services* dan *Regular Expressions* untuk menyediakan antarmuka verifikasi alamat yang terintegrasi dengan situs kodepos PT Pos Indonesia.

## 1.3. Tujuan dan Manfaat

Tujuan yang hendak dicapai dari tugas akhir ini yaitu menerapkan teknologi *Web Services* dan *Regular Expressions* untuk menyediakan fungsi verifikasi alamat yang mengacu pada data situs kodepos PT Pos Indonesia.

Sedangkan manfaat yang diharapkan dengan penyusunan tugas akhir yaitu untuk membantu proses verifikasi alamat pengiriman barang untuk berbagai situs klien.

## 2. Dasar Teori

### 2.1. Web Services

*Web Services* adalah berbagai jenis layanan aplikasi yang tersedia melalui jaringan internet, menggunakan sistem pengiriman pesan *eXtensible Markup Language* (XML) terstandarisasi dan tidak bergantung pada sistem operasi maupun bahasa pemrograman [2]. XML sendiri merupakan suatu format dokumen yang dirancang untuk mendeskripsikan struktur data dalam bahasa markah (*markup*). Dengan sifatnya yang tidak tergantung *platform* (sistem operasi dan bahasa pemrograman), *Web Services* mampu menjembatani proses pertukaran data antar berbagai perangkat yang berbeda *platform*.

Protokol *Web Services* terdiri atas empat lapisan yaitu [2]:

#### 1. *Service transport*

Lapisan ini bertugas untuk mengantarkan pesan antar aplikasi. Salah satu contoh protokol *service transport* yaitu HTTP (*HyperText Transfer Protocol*) yaitu protokol utama yang digunakan untuk pertukaran data antara *web browser* dengan *web server*.

#### 2. *XML Messaging*

Lapisan ini merupakan sistem perpesanan yang berfungsi menyatakan pesan ke dalam format dokumen XML. Sistem perpesanan yang digunakan antara lain SOAP (*Simple Object Access Protocol*) yaitu protokol pertukaran informasi antar komputer.

#### 3. *Service Description*

Lapisan ini bertugas mendeskripsikan antarmuka publik dari suatu *web services*. Lapisan ini ditangani oleh WSDL (*Web Services Description Language*) yaitu suatu tata bahasa XML yang menyatakan spesifikasi layanan *web services*.

#### 4. Service Discovery

Lapisan ini bertugas memusatkan layanan ke *registry* umum sebagai sarana publikasi dan pencarian.

### 2.2. Regular Expression

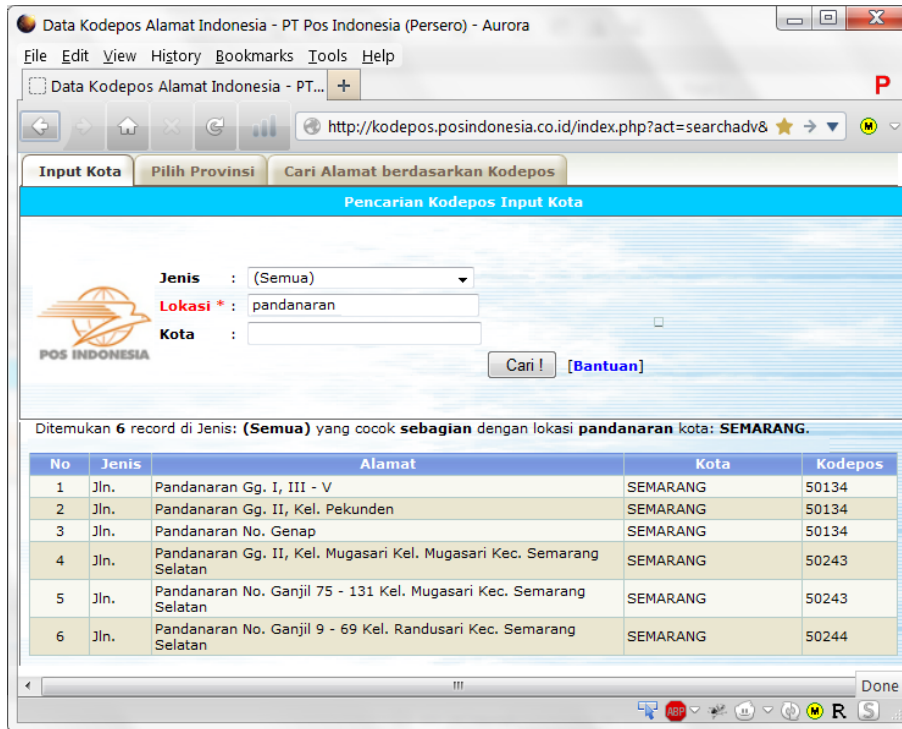
*Regular Expressions* (Regex) merupakan suatu metode yang sangat baik untuk memanipulasi data teks [6]. Regex merupakan solusi masalah pengolahan teks yang kompleks menjadi otomatis. Regex menggunakan beberapa simbol atau karakter khusus untuk menotasikan *pattern* (pola) yang terdiri atas meta-karakter dan repetisi. Meta-karakter merupakan simbol yang digunakan sebagai penanda *pattern*, sedangkan repetisi merupakan simbol yang menyatakan pengulangan *pattern* karakter. Regex didukung oleh bahasa skrip PHP melalui pustaka PCRE (*Perl-compatible Regular Expression*). Daftar meta-karakter Regex PHP ditunjukkan pada tabel 2.1.

Tabel 2.1. Meta-Karakter Regex pada PHP [1]

Simbol	Fungsi
/	mengawali dan mengakhiri (mengapit) <i>pattern</i>
^	mencocokkan <i>pattern</i> yang terletak pada awal subjek
\$	mencocokkan <i>pattern</i> yang terletak pada akhir subjek
.	mencocokkan dengan karakter apapun, kecuali baris baru
.*?	mencocokkan dengan karakter apapun termasuk baris baru
[ ]	membuka dan menutup definisi <i>character class</i>
	tanda pemisah dari untuk opsi alternatif
( )	membuka dan menutup sub- <i>pattern</i>
\	karakter <i>escape</i>
{x, y}	pembilang repetisi dengan nilai minimal x dan maksimal y
?	pembilang repetisi minimal nol dan maksimal satu {0, }
*	pembilang repetisi minimal nol dan maksimal tidak terbatas {0, 1}
+	pembilang repetisi minimal satu dan maksimal tidak terbatas {1, }

### 2.3. Situs Kodepos PT Pos Indonesia

Situs Kodepos PT Pos Indonesia (KPPI) merupakan salah satu layanan yang disajikan oleh situs PT Pos Indonesia [4]. Situs KPPI menyediakan fasilitas penelusuran data alamat dan kodepos berdasarkan *input* lokasi dan kota. Gambar 2.1 berikut ini merupakan contoh tampilan penelusuran pada situs KPPI.



Gambar 2.1. Screenshot Tampilan Penelusuran Situs KPPI

Tampilan penelusuran yang ditunjukkan pada gambar 2.1 di atas dibangun dari *sourcecode* HTML sebagaimana ditunjukkan pada gambar 2.2.

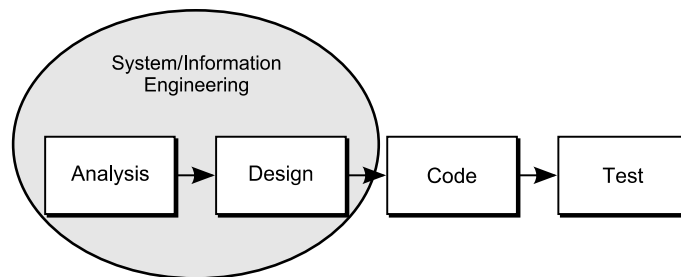


Gambar 2.2. Sourcecode Halaman Penelusuran KPPI

*Sourcecode* Halaman Penelusuran KPPI sebagaimana ditunjukkan oleh gambar 2.2 akan diuraikan (*parsing*) menggunakan Regex sehingga didapatkan datanya.

#### 2.4. Metode Rekayasa Perangkat Lunak

Rekayasa perangkat lunak adalah penerapan sistematis, disiplin, pendekatan kuantitatif terhadap pengembangan, operasi, dan pemeliharaan perangkat lunak [3]. Salah satu metode rekayasa perangkat lunak adalah model sekuensial linier yaitu pendekatan sistematis berurutan dalam pengembangan perangkat lunak yang dimulai pada tingkat sistem dan proses melalui tahap analisis, perancangan, implementasi, dan pengujian [3]. Model sekuensial linier terdiri atas beberapa tahapan sesuai yang ditunjukkan pada gambar 2.1 berikut.



Gambar 2.3. Model Sekuensial Linier [3]

Tahapan yang berlangsung pada model sekuensial linier sesuai gambar 2.1 yaitu:

1. *System/Information Engineering and Modelling*

Pekerjaan dimulai dari menetapkan kebutuhan untuk sistem elemen dan mengalokasikan kumpulan kebutuhan tersebut kepada perangkat lunak. Pandangan sistem menjadi penting tatkala perangkat lunak harus berinteraksi dengan elemen lain seperti perangkat keras, manusia, dan basis data.

2. *Analysis*

Tahap ini fokus pada menggambarkan apa yang dibutuhkan oleh pelanggan, membangun dasar bagi perancangan perangkat lunak (pemodelan data dan fungsi), dan membatasi serangkaian persyaratan yang dapat divalidasi saat perangkat lunak dibangun. Termasuk pula pendefinisian seluruh kebutuhan perangkat lunak, yang nantinya akan dijadikan sebagai *Software Requirements Specifications* (SRS). SRS merupakan dokumen yang berisi deskripsi lengkap mengenai apa kemampuan software tanpa menjelaskan bagaimana perangkat lunak akan melaksanakan kemampuan tersebut.

3. *Design*

Perancangan perangkat lunak merupakan proses multistep yang fokus pada atribut program: perancangan struktur data, antarmuka, dan fungsi atau algoritme.

#### 4. Code

Pada proses ini dilakukan pengkodean dari perancangan menjadi kode program yang dapat dibaca oleh mesin.

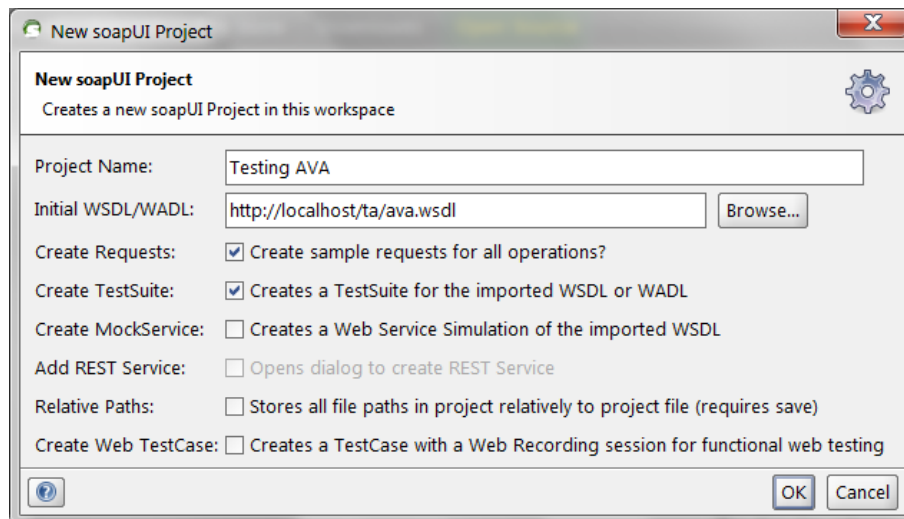
#### 5. Test

Saat kode program mulai dihasilkan, maka pengujian program dilangsungkan. Proses pengujian dilakukan untuk menemukan kekeliruan pemrograman dalam rangka memastikan *input* dapat menghasilkan *output* yang sesuai serta memastikan seluruh fungsi telah bekerja dengan benar.

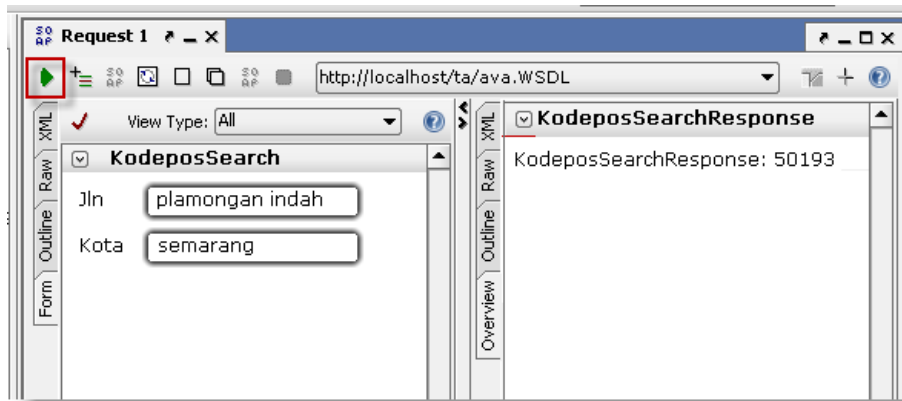
### 2.5. Pengujian dengan SoapUI

Perangkat lunak *Web Services* tidak memiliki tampilan antarmuka/form aplikasi, hanya menampilkan *output* data berformat XML. Agar pengujian *Web Services* dapat berlangsung secara interaktif, diperlukan perangkat lunak pihak ketiga yang berfungsi untuk menjabarkan *output* data berformat XML. Salah satu perangkat lunak dapat digunakan untuk menguji *Web Services* yaitu SoapUI. SoapUI merupakan perangkat lunak buatan SmartBear yang berfungsi untuk menguji fungsional perangkat lunak lainnya [5].

Gambar 2.4 dan 2.5 berikut ini masing-masing menunjukkan tampilan pembuatan project dan hasil eksekusi proses *request-response* pada SoapUI.



Gambar 2.4. Tampilan Pembuatan *Project* SoapUI



Gambar 2.5. Contoh Request-Response pada SoapUI

### 3. Analisis dan Perancangan Sistem

#### 3.1. Analisis

Deskripsi fungsional perangkat lunak Verifikasi Alamat (VA) terdapat pada tabel 3.1 berikut.

Tabel 3.1. SRS Verifikasi Alamat

Kode SRS	Deskripsi
SRS-F-01	Menampilkan dokumen WSDL
SRS-F-02	Verifikasi

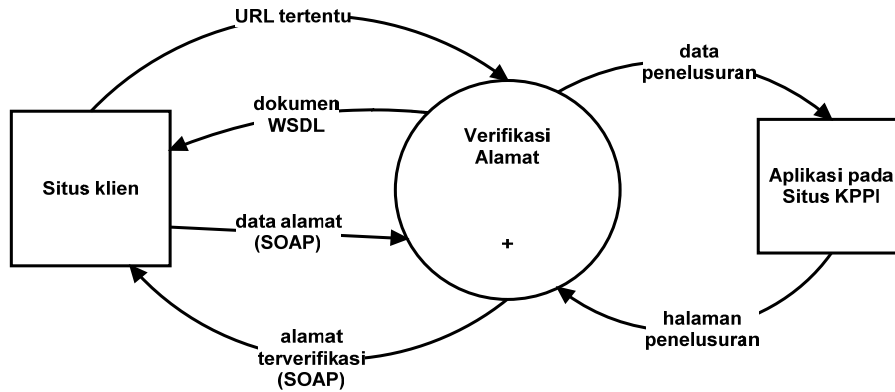
Pemodelan data VA dengan kamus data adalah sebagai berikut:

Data Alamat

type_alamat_output	=	type_alamat_input + {type_alamat_search} <sup>n</sup>
type_alamat_input	=	alamat_input + kabupaten_kota
type_alamat_search	=	alamat_search + kodepos + kelurahan + kecamatan
alamat_input	=	String
kabupaten_kota	=	String
alamat_search	=	String
kodepos	=	Integer
kelurahan	=	String
kecamatan	=	String

Pemodelan fungsi VA menggunakan Data Flow Diagram/Context Diagram ditunjukkan oleh gambar 3.1 berikut:





Gambar 3.1. DCD Verifikasi Alamat

### 3.2. Perancangan

Pemodelan data ditransformasikan menjadi menjadi struktur data bahasa pemrograman sebagaimana ditunjukkan pada tabel 3.2 berikut.

Tabel 3.2. Struktur Data Alamat

Indeks/ <i>field</i>	Tipe Data	Input/Output
alamat_input	<i>string</i>	Input-Output
kabkota	<i>string</i>	Input-Output
alamat_search	<i>string</i>	Output
kelurahan	<i>string</i>	Output
kecamatan	<i>string</i>	Output
kodepos	<i>integer</i>	Output

Pemodelan fungsional memetakan SRS dan DFD menjadi rincian fungsi sebagaimana ditunjukkan pada tabel 3.3.

Tabel 3.3. Pemodelan Fungsional

No.	Kode SRS	Deskripsi SRS	Nomor Fungsi	Nama Fungsi
1.	SRS-F-01	Menampilkan dokumen WSDL	FS-1	Spesifikasi Layanan
2.	SRS-F-02	Menerima <i>input</i> , memproses, dan menghasilkan <i>output</i> data alamat	FS-2 FS-2.1 FS-2.2 FS-2.3 FS-2.4	Verifikasi Menerima <i>Input</i> SOAP Penelusuran Penguraian Menghasilkan <i>Output</i> SOAP

Perancangan dokumen WSDL ditunjukkan pada tabel 3.4 berikut.

Tabel 3.4. Perancangan WSDL

Atribut Elemen	Nilai
definition.name	VerifikasiAlamat
definition.targetNamespace	urn:VerifikasiAlamat

Atribut Elemen	Nilai
types/element[0].name	TypeAlamatInput
types/element[0]/complexType/element[0].name	alamat_input
types/element[0]/complexType/element[1].name	kabkota
types/element[0]/complexType/element[0].type	<i>string</i>
types/element[0]/complexType/element[1].type	<i>string</i>
types/element[1].name	AlamatHasilSearch
types/element[1]/complexType/element[0].name	alamat_search
types/element[1]/complexType/element[1].name	kelurahan
types/element[1]/complexType/element[2].name	kecamatan
types/element[1]/complexType/element[3].name	kodepos
types/element[1]/complexType/element[0].type	<i>string</i>
types/element[1]/complexType/element[1].type	<i>string</i>
types/element[1]/complexType/element[2].type	<i>string</i>
types/element[1]/complexType/element[3].type	<i>integer</i>
types/element[2].name	TypeAlamatOutput
types/element[2]/complexType/element[0].name	InputPart
types/element[2]/complexType/element[1].name	SearchPart
types/element[2]/complexType/element[0].type	tns:TypeAlamatInput
types/element[2]/complexType/element[1].type	tns:TypeAlamatOutput
message[0].name	AlamatSearch
message[0]/part.name	InputAlamat
message[0]/part.type	tns:TypeAlamatInput
message[1].name	AlamatSearchResponse
message[1]/part.name	OutputAlamat
message[1]/part.type	tns:TypeAlamatOutput
portType.name	AlamatSearchPort
portType/operation.name	WS_AlamatSearch
portType/operation/input.message	tns:AlamatSearch
portType/operation/output.message	tns:AlamatSearchResponse
binding.name	AlamatSearchBinding
binding.type	tns:AlamatSearchPort
binding/operation.name	WS_AlamatSearch
binding/operation/input.namespace	urn:VerifikasiAlamat
binding/operation/output.namespace	urn:VerifikasiAlamat
service.name	VerifikasiAlamatService
service/port.name	AlamatSearchPort
service/port.binding	tns:AlamatSearchBinding
service/port/address.location	<a href="http://localhost/ta/VA.php">http://localhost/ta/VA.php</a>

## 4. Implementasi dan Pengujian

### 4.1. Implementasi

Sesuai perancangan fungsi dan WSDL, maka dihasilkan tiga berkas kode yaitu:

1. "verifikasiAlamat.wsdl", merupakan dokumen WSDL.
2. "va.php", merupakan target/letak *request* sesuai definisi pada dokumen WSDL

3. "function.php", berisi semua fungsional program yaitu:
  - a. "kppi\_curl\_search"  
 masukan : alamat dan kabkota (*array*)  
 keluaran : *sourcecode* HTML halaman penelusuran situs KPPI (*string*)
  - b. "parse\_html"  
 masukan : *sourcecode* HTML halaman penelusuran situs KPPI (*string*)  
 keluaran : alamat, kota, alamat verifikasi, kelurahan, kecamatan, dan kodepos (*array*)
  - c. "WS\_AlamatSearch"  
 masukan : alamat dan kabkota (*array*)  
 keluaran : dokumen SOAP sesuai dengan keluaran fungsi "parse\_html" (*string*)
  - d. "soap\_handle" (prosedur)  
 proses : menerima data SOAP dan memetakan fungsi WS\_AlamatSearch

## 4.2. Pengujian

Skenario pengujian terdiri atas kelas uji yang didasarkan pada SRS. Setiap kelas uji dibagi ke dalam beberapa nomor uji yaitu tindakan-tindakan yang dilakukan pada pengujian SRS yang bersangkutan. Tiap nomor uji akan dilakukan pengujian secara *black-box* dengan nilai tertentu untuk mengetahui ketepatan hasil dari masing-masing fungsi. Pemaparan skenario pengujian dapat dilihat pada tabel 4.1 berikut ini.

Tabel 4.1. Skenario Pengujian

No.	Kelas Uji / Kode SRS	Nomor Uji
1.	Menampilkan dokumen WSDL (SRS-F-01)	(U-01) Mengakses URL yang telah ditentukan (U-02) Mengakses sembarang URL
2.	Menerima <i>input</i> , memproses, dan menghasilkan <i>output</i> data alamat (SRS-F-02)	(U-03) Memasukkan data lokasi dan kota (U-04) Memasukkan data lokasi tanpa kota (U-05) Memasukkan data kota tanpa lokasi

Pengujian dinyatakan diterima apabila hasil yang diharapkan sesuai dengan hasil yang didapat untuk setiap nomor uji. Hasil uji terdapat ditunjukkan oleh tabel 4.2 dan 4.3 berikut ini.

Tabel 4.2. Hasil Uji SRS-F-01

Identifikasi dan Deskripsi	Mekanisme Pengujian	Masukan	Hasil yang Diharapkan	Hasil yang Didapatkan	Kesimpulan
(U-01) Mengakses URL yang telah ditentukan	<ol style="list-style-type: none"> <li>Membuat <i>project</i> baru di soapUI</li> <li>Memasukkan URL dalam <i>textbox</i> "Initial WSDL/WADL" pada tampilan <i>form</i> "New soapUI Project"</li> </ol>	URL <a href="http://localhost/ta/VerifikasiAlamat.wsdl">http://localhost/ta/VerifikasiAlamat.wsdl</a> atau <a href="file:///D:/xampp/htdocs/ta/VerifikasiAlamat.wsdl">file:///D:/xampp/htdocs/ta/VerifikasiAlamat.wsdl</a>	Tampilan <i>window</i> informasi <i>project</i> baru berisi definisi WSDL dan berbagai operasi yang disediakan oleh VA.	<i>Window</i> informasi definisi WSDL dan operasi tertampil.	Diterima
(U-02) Mengakses sembarang URL	<ol style="list-style-type: none"> <li>Membuat <i>project</i> baru di soapUI</li> <li>Memasukkan sembarang URL dalam <i>textbox</i> "Initial WSDL/ WADL" pada tampilan <i>form</i> "New soapUI Project"</li> </ol>	URL <a href="http://localhost/ta/VA.php">http://localhost/ta/VA.php</a>	Tampilan "Error loading WSDL"	"Error loading WSDL" tertampil	Diterima

Tabel 4.2. Hasil Uji SRS-F-02

Identifikasi dan Deskripsi	Mekanisme Pengujian	Masukan	Hasil yang Diharapkan	Hasil yang Didapatkan	Kesimpulan
(U-03) Memasukkan data alamat dan kota	Mengisikan masukan dalam tiap-tiap <i>textbox</i> Alamat dan Kota pada <i>window</i> "Request 1"	Alamat dan Kota	<i>Response</i> berupa kodepos sesuai masukan yang kemudian dibandingkan dengan penelusuran manual ke situs KPPI.	Keluaran sesuai	Diterima
(U-04) Memasukkan data alamat tanpa kota	Mengisikan masukan Kota pada <i>window</i> "Request 1"	Alamat	<i>SOAP Fault Error</i> ( <i>error</i> pada proses SOAP)	<i>SOAP Fault Error</i>	Diterima
(U-05) Memasukkan data alamat tanpa lokasi	Mengisikan masukan dalam tiap-tiap <i>textbox</i> Kota pada <i>window</i> "Request 1"	Kota	<i>SOAP Fault Error</i> ( <i>error</i> pada proses SOAP)	<i>SOAP Fault Error</i>	Diterima

Pada tabel 4.1 dan 4.2 ditunjukkan bahwa semua nomor uji telah teruji sesuai SRS.

Dengan demikian VA telah memenuhi untuk:

- Menghasilkan dokumen WSDL
- Melakukan verifikasi alamat

## 5. Kesimpulan dan Saran

### 5.1. Kesimpulan

Kesimpulan yang dapat diambil dari artikel ilmiah ini adalah sebagai berikut:

1. Dihasilkan perangkat lunak Verifikasi Alamat (VA) yang menerapkan teknologi *Web Services* dan *Regular Expressions*.
2. Melalui VA dimungkinkan proses pengaksesan data alamat pada situs KPPI oleh situs klien.
3. *Output* data alamat yang dihasilkan oleh VA dapat terdiri atas lebih dari satu baris sesuai hasil penelusuran situs KPPI. Dengan demikian, peran operator situs klien tetap dibutuhkan untuk memilih/mengoreksi data alamat tersebut.

### 5.2. Saran

Berikut ini adalah saran untuk pengembangan VA lebih lanjut:

1. Penambahan fungsi validasi pada *input* dari situs klien.
2. Penambahan fungsi seleksi otomatis atas *output* multi-baris.

## 6. Daftar Pustaka

- [1] Achour, Mehdi. dkk, 2011. "*PHP Manual*" diakses dari <http://php.net> pada September 2011
- [2] Cerami, Ethan, 2002, "*Web Services Essentials*", O'Reilly, Sebastopol California
- [3] Pressman, R.S., 2001, "*Software Engineering: A Practitioner Approach, Fifth Edition*", McGraw-Hill, New York City
- [4] PT Pos Indonesia, "*Sejarah PT Pos Indonesia (Persero)*", diakses dari <http://www.posindonesia.co.id/> pada 3 Juli 2012
- [5] SOAP-UI, "*What is SOAP-UI?*", diakses dari <http://soapui.org> pada Desember 2011.
- [6] Stubblebine, Tony, 2007, "*Regular Expression Pocket Reference, Second Edition*", O'Reilly, Sebastopol California
- [7] Yourdon, Edward, 2006, "*Just Enough Structured Analysis*". diakses dari <http://www.yourdon.com/jesa/> pada Januari 2011