

## INTEGRASI LAPORAN DEMAM BERDARAH *DENGUE* (DBD) MENGUNAKAN TEKNOLOGI *WEB SERVICE*

Ragil Saputra<sup>1)</sup>, Ahmad Ashari<sup>2)</sup>

1. Program Studi Teknik Informatika FMIPA Undip
2. Program Pasca Sarjana Ilmu Komputer FMIPA UGM

### Abstrak

Dengue Hemorrhagic Fever (DHF) is one of the infectious diseases that frequently leads to Extraordinary Situation. The management of report is conducted by Health Community Center which subsequently gives report to Health Office. A problem arising from the report management is the fact that the report is conducted manually, therefore, the data is less valid and is not processed as quickly as possible. The quick and accurate data report system enables to lessen the risk of Dengue Hemorrhagic Fever. Due to this fact, it is undeniable necessary to provide an integrated inter-system of Dengue Fever report. This system includes an inter-system between one Health Community Center to another and to the system in Health Office. The integration of inter-system report is able to be conducted by the use of web service technology. Therefore, this research focuses on the development of Web Service based integrated system on the report of Dengue Hemorrhagic Fever. Data exchange is conducted in XML form by the application of SOAP and WSDL technologies. Library NuSOAP is necessary to provide class soapClient and soapServer. In other words, it functions as the listener whose functions are to receive and to response at the access demand toward web service. The result is web service based report system which has dual functions since the system has functions to be either server or client.

**Key word :** web service, integration, SOAP, WSDL, DHF.

### 1. Pendahuluan

Perkembangan teknologi informasi semakin pesat hingga saat ini dengan terus dikembangkan teknologi-teknologi yang mendukungnya. Salah satu teknologi yang mendukung teknologi informasi adalah komputasi terdistribusi (*distributed computing*) yang memungkinkan dilakukan komputasi pada banyak mesin, dan hasilnya dimanfaatkan oleh banyak mesin pula. Salah satu teknologi komputasi terdistribusi yang berkembang pesat saat ini adalah *web service*. Konsep *web service* muncul untuk menjembatani sistem-sistem informasi yang ada tanpa mempermasalahkan perbedaan platform yang digunakan oleh masing-masing sumber [8].

*Web service* menyediakan standar komunikasi diantara berbagai aplikasi perangkat lunak yang berbeda-beda. *Web service* menggunakan teknologi komunikasi seperti SOAP (*Simple Object Application Protocol*), WSDL (*Web Service Description Language*) dan XML (*eXtensible Markup Language*) yang dikirim melalui HTTP (*Hypertext Transport Protocol*) [4]. Dengan penggunaan teknologi tersebut, *web service* menawarkan kemudahan untuk menjembatani pertukaran informasi yang digunakan. *Web service* merupakan turunan

aplikasi web dapat dibuat aplikasi modular yang dapat dipublikasikan, diletakkan, dan dibangkitkan antar web [8].

Dengan melihat keunggulan yang dimiliki *web service* tersebut, maka dalam penelitian ini diangkat topik mengenai penerapan *web service* pada pelaporan kasus penyakit Demam Berdarah *Dengue* (DBD) di Dinas Kesehatan Kabupaten Karanganyar.

Kabupaten Karanganyar merupakan daerah endemis DBD, ini terlihat dari survei lapangan oleh Dinkes Karanganyar terdapat lima kecamatan yang termasuk paling rawan serangan DBD karena angka bebas jentiknya hanya rata-rata 60% [1]. Idealnya, angka bebas jentik yang aman yakni antara 90% hingga 95%. Dan angka kesakitan (*incidence rate*) di Kabupaten Karanganyar masih cukup tinggi yaitu 5,7/10.000 penduduk (5,7 per 10.000 penduduk). Menurut Standart Pelayanan Minimal (SPM) nasional dalam penanggulangan DBD angka kesakitannya tidak boleh lebih dari 2/10.000 penduduk [11]. Berdasarkan data tersebut membuktikan bahwa penanggulangan DBD di Kabupaten Karanganyar belum optimal. Salah satu penyebabnya adalah dalam proses pelaporan kejadian DBD.

Berdasarkan uraian permasalahan tersebut, maka dapat disusun rumusan masalah yang akan dibahas dalam penelitian ini adalah :

- a. Merancang Sistem Pelaporan Demam Berdarah Dengue pada tiap-tiap Puskesmas untuk melakukan proses pencatatan dan Sistem Integrasi Data Laporan Demam Berdarah Dengue Dinkes untuk menerima laporan dari tiap-tiap puskesmas,
- b. Bagaimana mengintegrasikan sistem-sistem pada Puskesmas dan Dinkes yang masing-masing memiliki sistem dan basis data kedalam sistem terintegrasi menggunakan *web service*.

Hasil yang diperoleh penelitian ini adalah mengimplementasikan teknologi *web service* pada pencatatan dan pelaporan kasus Demam Berdarah Dengue (DBD) di Dinas Kesehatan Kabupaten Karanganyar.

## 2. Tinjauan Pustaka

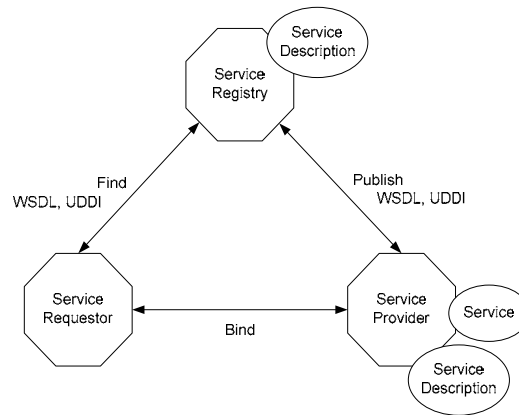
### 2.1. Web Service

Terdapat beberapa pengertian tentang *web service*, *web service* adalah antarmuka yang mendeskripsikan sekumpulan operasi yang dapat diakses dalam sebuah jaringan melalui pesan XML yang telah distandarkan [6]. W3C mendefinisikan *web service* sebagai sebuah sistem perangkat lunak yang dirancang untuk mendukung inter operasi mesin ke mesin di sebuah jaringan. *Web service* merupakan komponen perangkat lunak *loosely coupled*, dapat diguna ulang, membungkus fungsionalitas diskret, didistribusikan, dan diakses secara programatik melalui protokol internet standart [10].

Dari berbagai definisi tersebut dapat disimpulkan bahwa *web service* merupakan *middleware* internet yang memungkinkan berbagai sistem untuk saling berkomunikasi tanpa terpengaruh pada *platform*. *Web service* membungkus operasi-operasi ke dalam sebuah antarmuka yang ditulis dalam notasi XML. Antarmuka ini menyembunyikan detail implementasi dari layanan. Pertukaran informasi yang terjadi dalam *web service* juga menggunakan pesan dalam format XML.

*Web service* dibangun dari tiga komponen utama, yaitu *service provider*, *service registry*, dan *service requestor*. Komponen-komponen

tersebut saling berinteraksi melalui komponen *web service*, yang berupa deskripsi dan implementasi layanan. Terdapat tiga macam operasi yang memungkinkan komponen-komponen tersebut untuk dapat saling berinteraksi, yaitu *publish*, *find*, dan *bind*. Keterkaitan antara peran, operasi, dan komponen *web service* dapat dilihat pada Gambar 1.



Gambar 1. Komponen web service [6]

### 2.2. SOAP

SOAP (*Simple Object Acces Protocol*) adalah merupakan protokol untuk pertukaran informasi dengan desentralisasi dan terdistribusi [2]. SOAP merupakan gabungan antara HTTP dengan XML karena SOAP umumnya menggunakan protokol HTTP sebagai sarana *transport* datanya dan data akan dipertukarkan ditulis dalam format XML. SOAP mengatur bagaimana *request* dan *response* dari suatu *service* bekerja.

### 2.3. WSDL

WSDL (*Web Service Description Language*) adalah merupakan sebuah bahasa berbasis XML yang digunakan untuk mendefinisikan *web service* dan menggambarkan bagaimana cara untuk mengakses *web service* tersebut [3]. Fungsi utama WSDL dalam *web service* adalah untuk mengotomasi mekanisme komunikasi *business-to-business* dalam *web service* melalui protokol internet. Dengan menggunakan WSDL *client* dapat memanfaatkan fungsi-fungsi publik yang disediakan oleh server [8].

## 2.4. PHP NuSOAP

NuSOAP adalah sebuah kumpulan *class-class* PHP yang memungkinkan user untuk mengirim dan menerima pesan SOAP melalui protokol HTTP [9]. NuSOAP merupakan *toolkit web service* berbasis komponen. NuSOAP memiliki beberapa *class* yang menyediakan *method* seperti serialisasi variabel dan pemaketan *SOAP-Envelope*. Interaksi *web service* dilakukan dengan *class client* yang disebut dengan *class soapClient* dan *class server* yang disebut dengan *class soapServer*. *Class-class* ini mengizinkan *user* untuk melakukan proses pengiriman dan penerimaan pesan-pesan SOAP dengan bantuan beberapa *class-class* pendukung lainnya untuk melengkapi proses tersebut.

## 3. Analisa Dan Rancangan

Sistem *web service* pelaporan Demam Berdarah Dengue (DBD) yang di bangun merupakan sistem pelaporan yang menggunakan *web service*, yang melibatkan sistem Dinas Kesehatan dan seluruh sistem DBD Puskesmas di wilayah kabupaten Karanganyar. Adapun analisa kebutuhan sistem antara lain [7]:

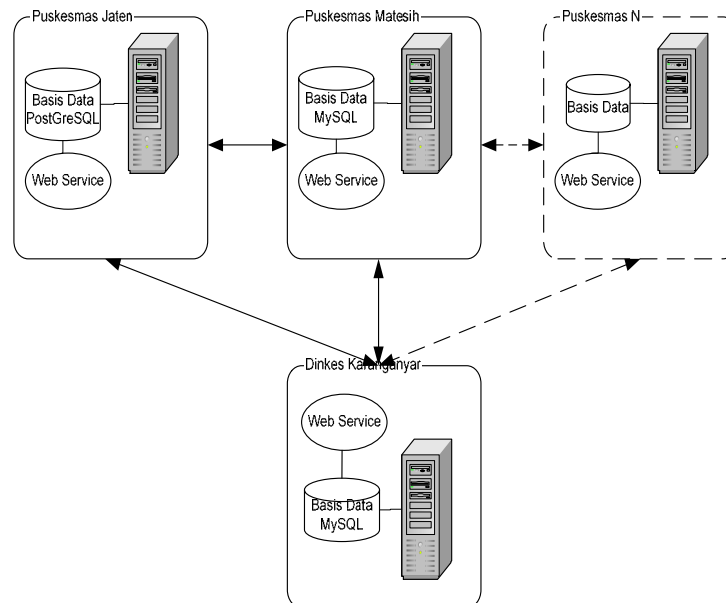
- Dibutuhkan satu layanan *web service* yang terdiri dari dua bagian. Bagian yang pertama merupakan *web service server* di tiap unit

puskesmas. Setiap *web service server* berfungsi sebagai *provider* data dari masing-masing puskesmas. Bagian kedua merupakan *web service* klien yang akan mengambil data untuk diintegrasikan. Dalam hal ini dinkes sebagai klien untuk merequest laporan kejadian. Serta berlaku sebaliknya, dinas kesehatan menjadi server yang menyediakan info/berita yang dapat diakses seluruh sistem puskesmas yang terhubung.

- Dibutuhkan layanan sebuah sistem integrasi data laporan DBD pada dinas kesehatan untuk menginformasikan atau menampilkan data maupun laporan dari masing-masing puskesmas, serta penyampaian informasi dari dinas kesehatan keseluruhan puskesmas.

## 3.1. Arsitektur Sistem

Pada penelitian ini dibuat sebuah rancangan model arsitektur sistem pelaporan DBD berbasis *web service*. Pengembangan aplikasi sistem berbasis model *n-tier* karena model tersebut menggunakan *web service* sebagai media komunikasi pada database pada masing-masing puskesmas dan dinkes. Model rancangan arsitektur umum dapat disajikan pada Gambar 2. Pada model tersebut dapat dilakukan penambahan sistem puskesmas sampai dengan N buah.



Gambar 2. Arsitektur sistem

### 3.2. Sistem Puskesmas

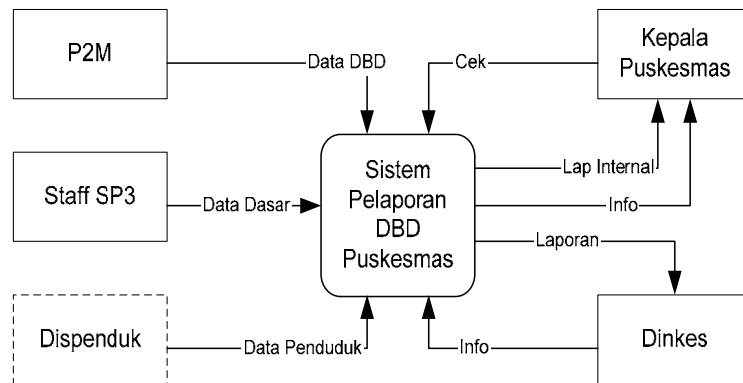
Sistem puskesmas melakukan pencatatan kejadian Demam Berdarah Dengue dan melaporkan ke dinkes berdasarkan form KD/RS (formulir Kewaspadaan Dini/Rumah Sakit) yang berisi data pasien dan hasil pemeriksaan pasien, serta kondisi pasien. Adapun aktor/pengguna yang terlibat dalam sistem pelaporan DBD puskesmas antara lain :

- Koordinator P2M, bertugas menangani data DBD dan membuat laporan.
- Kepala Puskesmas, merupakan pengguna laporan ditingkat Puskesmas dan

penanggungjawab data laporan ke Dinas Kesehatan.

- Pengelola SP3, merupakan petugas yang melakukan entri data dasar sistem.
- Dinkes, sistem dinkes merupakan entitas eksternal dari sistem puskesmas yang terhubung melalui *web service*.

Untuk kelengkapan sistem diperlukan data jumlah penduduk per desa, data ini di dapat dari Dinas Kependudukan (Dispenduk). Diagram konteks sistem puskesmas dapat disajikan pada Gambar 3.



Gambar 3. DCD sistem puskesmas

### 3.2. Sistem Dinkes

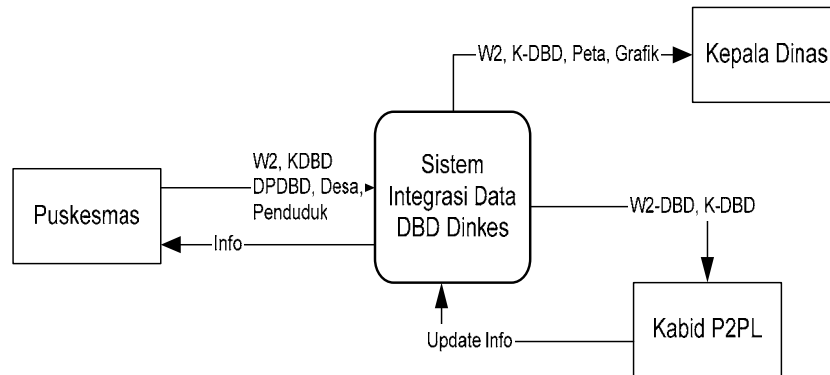
Sistem dinkes secara umum merupakan client yang meminta layanan kepada sistem puskesmas. Sistem integrasi data laporan DBD di dinas kesehatan melibatkan beberapa aktor sebagai berikut :

- Kepala Bidang Kabid P2PL (Pengendalian Penyakit dan Penyehatan Lingkungan), bertanggungjawab menangani laporan kasus DBD dari tingkat Puskesmas, menerima laporan data pasien, dan melakukan *update* berita.

- Kepala Dinas Kesehatan, merupakan aktor penanggungjawab pelaporan ke instansi lain, dan dapat melihat hasil seluruh laporan dalam bentuk tabel, grafik dan peta.

- Puskesmas, sistem puskesmas merupakan entitas eksternal yang terhubung melalui *web service* yang menyediakan data.

Keterlibatan masing-masing user dan sistem dapat disajikan dalam diagram konteks, berikut ini diagram kontek sistem dinkes dapat disajikan pada Gambar 4.



Gambar 4. DCD sistem dinkes

#### 4. Implementasi

Pada tahap implementasi dibuat beberapa fungsi atau *service*. Pembuatan fungsi ini dibagi menjadi beberapa fungsi, yang masing-masing fungsi memiliki parameter *input* dan *output* yang berbeda sesuai dengan kebutuhan transaksi data. Pembagian fungsi bertujuan untuk efisiensi kinerja dari proses pelaporan dan kemudahan melakukan *debugging* dari proses integrasi sistem ini. Adapun *service* yang digunakan antara lain sebagai berikut [7] :

a. *getLaporanW2*

*Service* ini yang digunakan untuk mendapatkan dan mengirimkan laporan W2 atau laporan mingguan. *Service* ini memiliki parameter *input* berupa minggu, tahun kejadian dan tahun penduduk. Sedangkan outputnya berupa nama desa, jumlah kejadian, jumlah kematian, dan jumlah penduduk.

b. *getLw2\_Puskesmas*

*Service* ini digunakan untuk mengirimkan laporan W2 semua kejadian per puskesmas per minggu. *Service* ini memiliki parameter *input* tahun kejadian dan minggu. Sedangkan output berupa nama puskesmas dan total penderita.

c. *getLw2\_puskesmasAll*

*Service* ini digunakan untuk mengirimkan laporan W2 semua kejadian per puskesmas sampai dengan minggu terakhir. *Service* ini memiliki parameter *input* tahun kejadian dan minggu. Dan output berupa nama puskesmas, jumlah penderita berdasarkan minggu dan total penderita.

d. *getDataPasien*

Merupakan *service* untuk mendapatkan data detail pasien, dengan parameter *input* berupa munggu, tahun kejadian, puskesmas dan desa. Sedangkan output yang dihasilkan berupa array data pasien yaitu nama, umur, alamat, status pasien.

e. *getTotalPenderita*

*Service* ini digunakan untuk menghitung total penderita dari seluruh desa. Parameter *input* berupa tahun kejadian, dan outputnya total semua kejadian pada puskesmas tersebut.

f. *getStatus*

Merupakan *service* untuk mengetahui jumlah pasien dengan status terakhir pasien hidup atau meninggal. Parameter *input* berupa tahun, output berupa total pasien yang hidup atau meninggal.

g. *getPenderitaFPeta*

Merupakan *service* untuk mendapatkan warna peta desa, berdasarkan nilai IR tiap desa, dengan parameter *input* id\_peta dan output berupa warna peta desa.

Implementasi web service pada sistem ini dikembangkan menggunakan library NuSoap. Library ini menyediakan kelas-kelas yang dapat digunakan untuk membuat web service, diantaranya *soapServer* dan *soapClient*. *SoapServer* dapat digunakan untuk membuat *instance server* web service yang berfungsi untuk menyediakan web service (*service provider*) atau sebagai *listener* yang akan menerima dan merespon permintaan akses terhadap web service. Sedangkan *soapClient* digunakan untuk membuat *instance client* web service yang berfungsi untuk menggunakan

layanan web service yang ada atau sebagai *service requestor*.

Pembuatan sebuah *service* dimulai dengan memanggil library NuSOAP :

```
require_once('nusoap.php');
```

Kemudian membuat objek *soap\_server* :

```
$server = new soap_server();
```

Selanjutnya fungsi tersebut diberi nama atau *namespace* untuk ditambahkan sebagai informasi dalam dokumen WSDL. *Namespace* adalah sebuah nama unik yang diberikan kepada *web service*.

```
$server->configureWSDL('puskesmasserver',  
'urn:puskesmasserver');
```

Setiap *web service* harus memiliki *namespace* yang berbeda karena dalam penerapannya ada kemungkinan *web service* yang ada saling berinteraksi dan memerlukan struktur data atau bahkan fungsi *web service* lain.

Berikutnya fungsi tersebut didaftarkan kedalam *web service* agar menjadi *method* dari *service*. Proses pendaftaran juga menyertakan tipe data dan nama *service* yang digunakan serta URN (*Uniform Resource Name*) yang

digunakan. Pada pendaftaran tipe data, jika berupa *complexType* harus didefinisikan satu persatu tipe datanya dan schema masing-masing data.

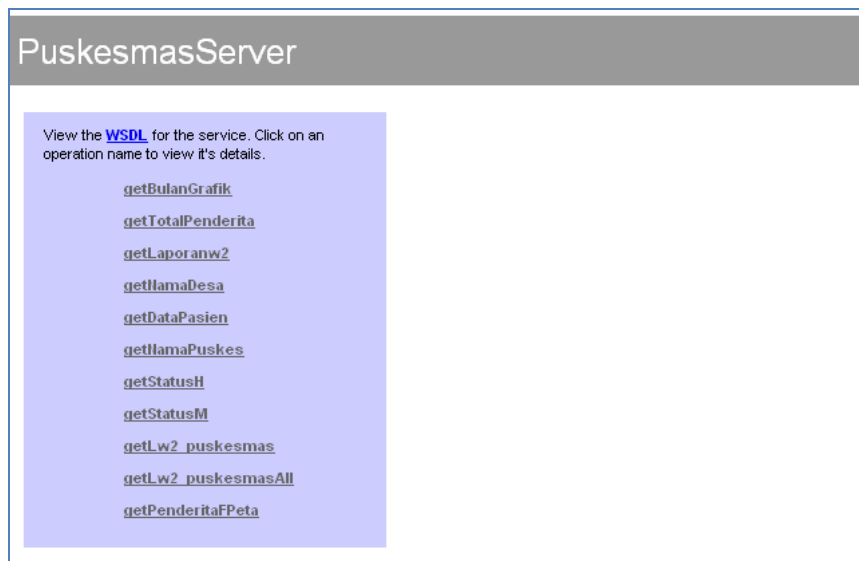
```
$server->WSDL_addComplexType
```

```
$server->register("getLaporanW2", array  
("minggu"=>"xsd:int"));
```

Langkah selanjutnya yaitu mendefinisikan fungsi, pada proses ini dilakukan *query* ke basis data berdasarkan parameter yang dimasukkan ke fungsi tersebut. Dan hasil *query* disimpan dalam sebuah variabel array sesuai dengan *type* data yang telah didefinisikan.

Tahap terakhir adalah pengecekan terhadap isi variabel `$HTTP_RAW_POST_DATA`. Jika belum terbentuk maka variabel diisi dengan variabel kosong. Isi variabel tersebut kemudian akan diberikan kepada *web service* untuk diproses.

Hasil pembuatan *service server* disimpan pada lokasi server. Puskesmas Jatipuro `http://172.16.1.10/puskesmas_service.php`, *service* yang disediakan terdapat pada Gambar 5.



Gambar 5. Daftar Web Service Puskesmas

Untuk mengakses *service server* puskesmas Jatipuro digunakan *web service client* dinkes dengan membuat *instance* client dinkes :

```
$client = new soapclient(http://  
172.16.1.10/puskesmas_service.php);
```

Kemudian melakukan pemanggilan SOAP *methode* untuk mendapatkan data laporan. Misalnya untuk mendapat laporan W2 (mingguan), dapat dilakukan akses terhadap *service* `getLaporanW2` sebagai berikut :

```
$result=$client -> call( "getLaporanw2",array(
"minggu"=>$mingguke);
```

Berdasarkan Gambar 2 Arsitektur sistem, maka dengan cara yang sama dibuat *web service server* pada tiga puskesmas yang berbeda, dan dilakukan pengaksesan dari *web service client* dinkes ke *web service* puskesmas tersebut.

### 5. Hasil dan Pengujian

Berdasarkan arsitektur sistem pada Gambar 2, dilakukan pengujian dengan menggunakan empat buah komputer, yang bertindak sebagai sistem puskesmas (tiga komputer) dan sebuah komputer bertindak sebagai sistem dinkes. Daftar alamat IP dapat disajikan pada Tabel 1.

Table 1 Tabel alamat IP pengujian sistem

Alamat IP	Keterangan
172.16.1.10	Sistem puskesmas jatipuro
172.16.1.11	Sistem Dinkes
172.16.1.13	Sistem puskesmas matesih
172.16.1.14	Sistem puskesmas jaten

#### 5.1 Sistem Puskesmas

Pengujian sistem puskesmas dimulai dengan proses entri data pasien oleh petugas P2M. Data pasien bersifat umum belum ada laporan kejadian DBD, karena puskesmas baru melakukan pemeriksaan dan pasien di duga (*suspect*) DBD. *Update* data pasien dilakukan setelah puskesmas mendapat tembusan laporan form KD/RS dari rumah sakit yang melakukan pemeriksaan. Gambar 6 merupakan formulir untuk entri data KD/RS.

Gambar 6. Menu entri KD/RS

Proses entri data KD/RS atau data pasien pada salah satu sistem dipuskesmas akan menyebabkan jumlah data kejadian pada saat tersimpan akan dilakukan *update* (perubahan), sehingga semua sistem baik dinkes maupun puskesmas lain yang melakukan akses melalui *web service* ke sistem tersebut juga akan

menampilkan secara *real time*. Hal ini dapat dilakukan karena setiap sistem disediakan sebuah mekanisme untuk melakukan *recall* (pemanggilan berulang-ulang) terhadap *service* dengan cara melakukan *refresh* browser secara berkala.

### 5.2 Sistem Dinas Kesehatan

Pengujian selanjutnya dilakukan pada sistem dinas keseshatan, pada Gambar 7 merupakan

laporan w2 mingguan per desa, laporan mingguan yang dapat diakses oleh kepala dinas.

(Pemantauan Kasus Pada Mingguke 37 Tahun 2010)							
NAMA PUSKESMAS	NO	DESA	JUMLAH PENDUDUK	JUMLAH		IR	CFR
				P	M		
Jatipuro	1	Ngepungsari	3298	0	0	0	0 %
	2	Jatisuko	3200	0	0	0	0 %
	3	Jatipuro	4012	0	0	0	0 %
	4	Jatimulyo	6320	0	0	0	0 %
	5	Jatikuwung	4312	0	0	0	0 %
	6	Jatiharjo	5278	0	0	0	0 %
	7	Jatipurwo	4986	0	0	0	0 %
	8	Jatiwarno	3987	0	0	0	0 %
	9	Jatiroyo	6190	0	0	0	0 %
	10	Jatisobo	4765	0	0	0	0 %

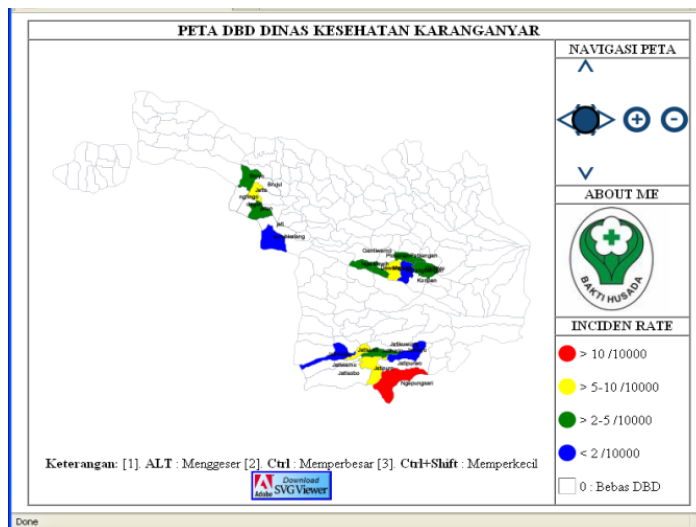
Gambar 7. Halaman laporan W2 per desa

Pada halaman menu Gambar 7 dapat dibuat laporan KDBD per desa atau laporan tahunan, dengan cara melakukan seleksi pemilihan minggu=ALL.

Penyajian laporan dalam bentuk visualisasi sangat membantu dalam membaca data, user dapat memilih menu peta endemis, sistem akan merespon dengan menampilkan *window browser* baru yang memuat halaman peta. *Service* yang menangani *request* dari sistem dinkes dan menyediakan warna peta desa adalah *service*

*getPenderitaFPeta*. Setelah berhasil didapatkan nilai warna peta, langkah selanjutnya yaitu menggabungkan dengan data peta yang disimpan pada sistem dinkes.

Pada pengujian sistem ini digunakan beberapa browser untuk dilakukan pengujian. Pada Gambar 8 adalah hasil visualisasi peta berdasarkan nilai IR (*Incidence Rate*) masing-masing desa dengan browser mozilla firefox yang sudah ditambahkan plugin SVG viewer.

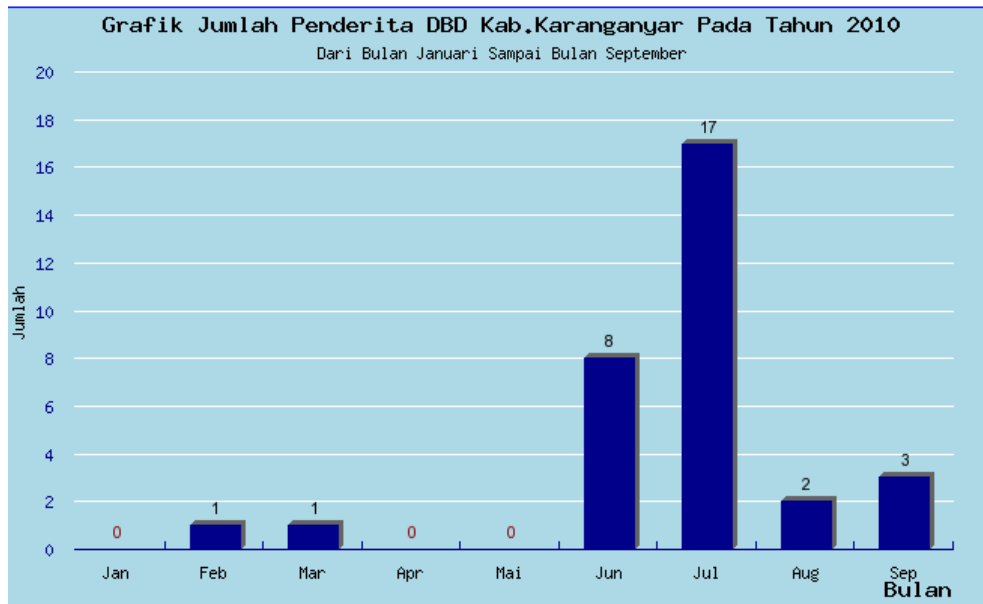


Gambar 8. Peta DBD



Laporan visual yang lain adalah grafik, user disediakan pilihan untuk membuat grafik berdasarkan bulan. Service yang dipanggil adalah *service getTotalPenderita* kemasing-masing puskesmas kemudian dijumlahkan. Pada

sistem akan ditampilkan window baru yang menampilkan grafik trend bulanan. Pada Gambar 9 merupakan hasil jumlah penderita per bulan.

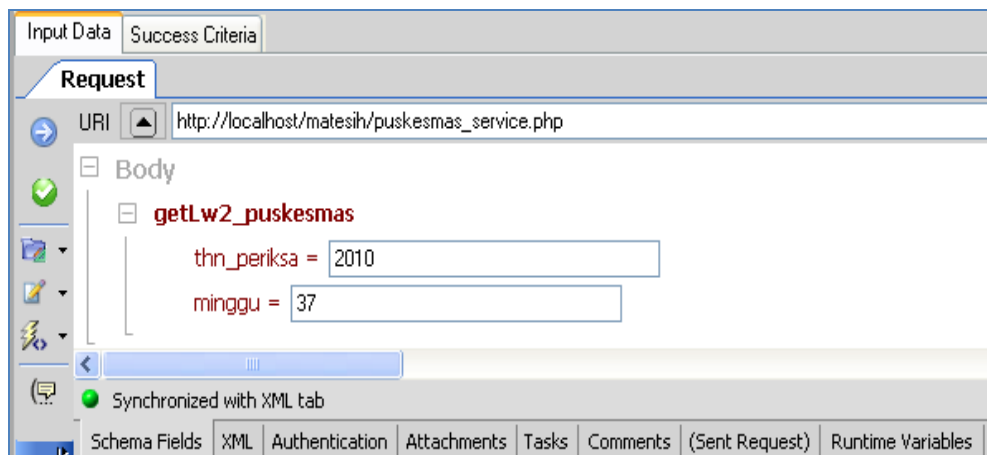


Gambar 9. Grafik DBD

### 5.3 Pengujian Fungsional Sistem

Pengujian fungsional sistem menggunakan *automation tool* SOAPSonar yang mempunyai kemampuan menguji *white box* melalui file WSDL yang merupakan model *web service* dari masing-masing sistem. Sebagai contoh

dilakukan pengujian terhadap *service getLw2\_puskesmas*, untuk mendapatkan jumlah kejadian berdasarkan minggu. Parameter yang diujikan adalah tahun kejadian dan minggu ke, hasil capture untuk *input* parameter dapat dilihat pada Gambar 10.



Gambar 10. Input parameter

Pada Gambar 10 dimasukkan parameter uji adalah tahun = 2010 dan minggu = 37, pesan SOAP *request* yang dihasilkan dalam format XML terlihat pada Gambar 11.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope>
...
<soap:Body soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<tns:getLw2_puskesmas>
<thn_periksa xsi:type="xsd:string"> 2010
</thn_periksa>
<minggu xsi:type="xsd:int">37</minggu>
</tns:getLw2_puskesmas>
</soap:Body>
</soap:Envelope>
```

Gambar 11. Pesan SOAP (Sent Request)

Pesan SOAP ini dikirimkan ke server puskesmas, dengan target name space atau *tns* : *getLw2\_puskesmas*. Kemudian server puskesmas mengembalikan nilai melalui *Soap response* dari server puskesmas, dan hasilnya terdapat pada tampilan Gambar 12.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
...
<SOAP-ENV:Body>
<ns1:getLw2_puskesmasResponse
xmlns:ns1="urn:PuskesmasService">
<return xsi:type="xsd:int">1</return>
</ns1:getLw2_puskesmasResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Gambar 12. Pesan SOAP (Response)

Nilai yang dikembalikan adalah **1** dengan type integer. Dengan demikian pada minggu ke **37** tahun **2010** pada puskesmas tersebut terdapat **1** kejadian.

## 6. Kesimpulan

Berdasarkan pemaparan dari proses desain, implementasi, hasil dan pengujian dapat diperoleh beberapa kesimpulan :

a. Sistem integrasi data laporan Demam Berdarah Dengue (DBD) dinas kesehatan

Karanganyar berhasil diimplementasikan menggunakan teknologi *web service*.

b. Pengujian interoperabilitas antara sistem puskesmas dengan sistem dinkes dilakukan pada mesin yang berbeda dengan *platform* basis data MySQL dan PostgreSQL.

## Referensi

- [1] Anonim, *Angka bebas jentik hanya 60% 5 Kecamatan rawan serangan DBD*. www.karanganyar.go.id, di akses 24 November 2009.
- [2] Box, D., Ehnebuske, D., Kakivaya, G., Layman, A., Mendelsohn, N., Nielsen, H.F., Thatte, S. and Winer, D., *Simple Object Acces Protocol (SOAP) 1.1*, 2000 <http://www.w3c.org/TR/2000/NOTE-SOAP-20000508/> diakses 29 Januari 2010.
- [3] Christensen, E., Curbera, F., Meredith, G. and Weerawarana, S., *Web Services Description Language (WSDL) 1.1*, 2001 <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>, diakses 6 Februari 2010.
- [4] Goottschalk, K., *Web Service Architecture*, 2002, www.pa.icar.cnr.it/.../D30.2%20%20Introduction%20to%20Web%20Service%20Architecture.pdf diakses 29 Januari 2010.
- [5] Jorgensen, D., *Developing .Net Web Service with XML*, Syngress, 2002.
- [6] Kreger, H., *Web Service Conceptual Architecture (WSA 1.0)*, 2001, [www.cs.uoi.gr/~zarras/mdw.../WebServicesConceptualArchitectu2.pdf](http://www.cs.uoi.gr/~zarras/mdw.../WebServicesConceptualArchitectu2.pdf) diakses 6 Februari 2010.
- [7] Saputra, R., *Implementasi Web Service pada Pelaporan Demam Berdarah Dengue (DBD) di Dinkes Karanganyar*, Tesis, Universitas Gadjah Mada, Yogyakarta, 2010.
- [8] Siswoutomo, W., *Membangun Web Service Open Source Menggunakan PHP*, Elex Media Komputindo, Jakarta, 2004.
- [9] Stogov, D., *PHP SOAP extension*, 2004. <http://devzone.zend.com/article/689-PHP-SOAP-Extension>, diakses 6 Februari 2010.
- [10] Systinet, *Web Services : A Practical Introduction*, The Stencil Group, 2003.
- [11] Warsito, *Rencana Kerja Peningkatan Kemampuan Pegawai dalam*

*Penanggulangan Penyakit DBD pada Seksi  
Pemberantasan dan Pengendalian Penyakit  
Dinas Kesehatan Kabupaten Karanganyar,*

Kertas Kerja Perseorangan, Badan Diklat  
Provinsi Jawa Tengah, Semarang, 2009

