

Implementasi Web Services Pada Sistem Integrasi Data Laporan Bulanan Penyakit Dari Puskesmas-Puskesmas

Supriyono

Prodi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Cirebon
neopriyo@gmail.com

Abstrak

Sistem basis data yang tersebar pada tiap puskesmas menyulitkan *user* untuk melihat data-data laporan penyakit secara *real*. Seluruh basis data tersebut diintegrasikan dengan sebuah sistem manajemen basis data terintegrasi melalui mekanisme sistem berbasis *web client-server*. Sistem pada Dinas Kesehatan memiliki *web service* yang berfungsi sebagai *server* untuk layanan permintaan dari puskesmas. Implementasi *web service* tersebut termasuk *business-service tier*, artinya *web service* tidak dibuat untuk berinteraksi langsung dengan *user*. Peran SOAP adalah sebagai *protocol XML based* untuk pertukaran informasi antar komputer melalui protokol *transport*.

Prototype sistem ini mengolah daftar objek data puskesmas, jenis penyakit, penyakit, pasien, periksa, laporan dan kriteria umur pada setiap *server* puskesmas yang menyediakan URL *service provider* dan *service requestor*. Teknologi *web service* memungkinkan penggabungan aplikasi layanan dari sistem yang berada pada masing-masing puskesmas di lingkungan dinas kesehatan Kabupaten Cirebon.

Penerapan *web service* pada replikasi data di sisi *server* akan meningkatkan kecepatan pada saat diakses. Hasil penelitian *web service* ini merupakan infrastruktur yang baik untuk integrasi data penyakit yang tersebar di *server-server* puskesmas. Integrasi data yang dibangun menggunakan *web service* pada prototip sistem integrasi data dapat dimanfaatkan sebagai sumber data dan informasi untuk mendukung dan meningkatkan kinerja pemerintah daerah khususnya Dinas Kesehatan.

Kata kunci: administrasi, penyakit, integrasi, data, *web service*, SOAP.

1. PENDAHULUAN

Pengolahan data dan penyebaran informasi menjadi kurang efektif dan efisien, apabila sumber informasi dalam bentuk kertas yang statis atau mengandalkan daya ingat seseorang sebagai media penyimpanannya. Dinas Kesehatan Kabupaten Cirebon dalam melakukan administrasi dan pengolahan data laporan bulanan penyakit dari puskesmas-puskesmas masih dilakukan secara manual. Pengolahan data secara manual tersebut sering menimbulkan masalah antara lain: kehilangan data, laporan yang tidak aktual dan akurat.

Permasalahan tersebut menimbulkan kesulitan puskesmas-puskesmas saat

melakukan pelaporan. Permasalahan dalam pelaporan biasanya muncul baik terhadap pimpinan maupun bagian lain yang terkait laporan data penyakit. Laporan yang disampaikan sering terlambat dan berubah-ubah datanya karena harus menunggu rekap dari dari petugas yang menyusun laporan dan pengisian laporannya sendiri juga membutuhkan waktu.

Dalam rangka dan meningkatkan efektifitas dan efisiensi dalam pelaporan bulanan penyakit dari puskesmas kepada Dinas Kesehatan Kabupaten Cirebon diperlukan suatu sistem informasi berbasis *web* agar laporan penyakit tersebut dapat dilakukan dengan lebih mudah, cepat dan akurat.

PHP *web services* adalah aplikasi yang dibuat agar dapat diakses oleh aplikasi lain melalui *internet* maupun *intranet* dengan menggunakan XML sebagai format pengiriman pesan. Sistem yang menggunakan PHP sebagai format dokumennya memungkinkan *web services* dalam berkomunikasi antar aplikasi dan *platform* yang berbeda. Konsep teknologi *web service* muncul untuk mendukung sistem terdistribusi yang berjalan pada infrastruktur yang berbeda. SOAP (*Simple Object Access Protocol*) dan beberapa teknologi yang didukung seperti WSDL (*Web Services Description Language*) dan UDDI (*Universal Description Discovery and Integration*) merupakan kombinasi dari XML (*eXtensible Markup Language*) yang dikirimkan melalui HTTP (*HyperText Transfer Protocol*).

2. METODE PENELITIAN

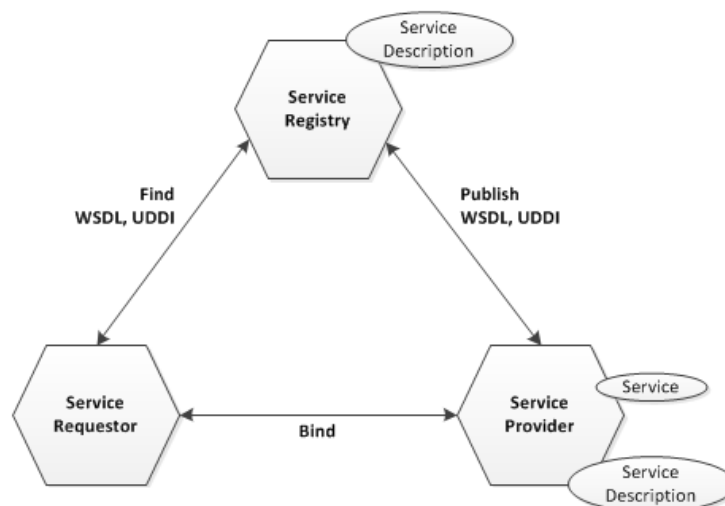
2.1. Web Service

Web service adalah seperangkat protokol komunikasi yang terdiri dari tiga komponen. Komponen dalam *web service*, yaitu *Simple Object Application Protocol* (SOAP), *Universal Description Discovery and Integration* (UDDI) yang merupakan kombinasi dari *eXtensible Markup Language* (XML), dan *Web Service Description Language* (WSDL) yang dikirimkan melalui HTTP [1].

Web service sangat potensial bagi perkembangan kolaborasi aplikasi B2B (*Business to Business*). Jika Dua buah perusahaan ingin saling bertukar informasi dapat dilakukan melalui proses yang panjang dan melelahkan. Sebagai contoh, mengatur sistem inventori suatu perusahaan agar dapat berkomunikasi langsung dengan sistem produksi *supplier* pasti melalui proses negoisasi yang panjang, bagaimana memanggil suatu fungsi, seperti apa format dokumen yang akan dipertukarkan dan seterusnya [2].

Web service merupakan sistem yang berbentuk *Service Oriented Architecture* (SOA) yang berdasarkan pada interaksi yaitu Tiga peran seperti yang diperlihatkan pada Gambar 1 yaitu:

1. *Service Provider*: merupakan pemilik *web service* yang berfungsi untuk menyediakan layanan (kumpulan operasi) dari *web service* dan mengelola *register* agar layanan-layanan tersebut dapat tersebut dapat tersedia.
2. *Service Registry*: berfungsi sebagai lokasi sentral yang mendeskripsikan semua layanan yang teregister dan sebagai tempat tempat *service provider* mempublikasikan layanannya.
3. *Service Requestor*: merupakan aplikasi yang bertindak sebagai klien dari *web service* yang mencari dan meminta layanan yang dibutuhkan serta menggunakan layanan tersebut.



Gambar 1. Arsitektur *web service* [1]

Tiga operasi yang dilakukan *web service* [3], yaitu:

1. *Publish/Unpublish*: menerbitkan/menghapus layanan ke dalam atau dari suatu *registry*. *Service provider* menghubungi *service broker* untuk *publish* atau *unpublish* suatu layanan.
2. *Find* yaitu operasi yang dilakukan secara bersama-sama antara *service requestor* dan *service provider*. *Service requestor* mencari dan menemukan layanan yang dibutuhkan, dan *service broker* memberikan layanan yang diinginkan.
3. *Bind*: operasi *bind* terjadi antara *service requestor* dan *service provider*. Setelah *service requestor* menemukan layanan yang dicari, kemudian melakukan *binding* ke *service provider* untuk melakukan interaksi dan mengakses layanan yang disediakan oleh *service provider*.

2.2. XML (*eXtensible Markup Language*)

XML merupakan dasar terbentuknya *web service* yang digunakan untuk mendeskripsikan data. Pada level paling detail *web service* secara keseluruhan dibentuk di atas XML. Fungsi utama dari XML adalah komunikasi antar aplikasi, integrasi data, dan komunikasi antar aplikasi eksternal dengan *partner* luar. Dengan standarisasi XML, aplikasi-aplikasi yang berbeda dapat dengan mudah berkomunikasi antar satu dengan yang lain [4].

XML merupakan bahasa *markup* yang digunakan untuk mengolah *meta data* (informasi tentang data) yang menggambarkan struktur dan maksud/tujuan data yang terdapat dalam dokumen XML, namun, bukan menggambarkan format tampilan data tersebut. XML dirancang untuk membawa data, mendeskripsikan data dan berfokus

apakah data itu. XML merupakan sebuah standar sederhana yang digunakan untuk mendeskripsikan data teks dengan cara *self-describing* (deskripsi diri). XML juga merupakan *markup language* untuk dokumentasi terstruktur. Dokumen-dokumen terstruktur adalah dokumen-dokumen yang mempunyai *content* (kata, gambar) serta indikasi yang menyatakan makna dari *content* tersebut.

2.3. SOAP (*Simple Object Access Protocol*)

SOAP merupakan protokol untuk pertukaran informasi dalam format XML. SOAP merupakan gabungan antara HTTP dengan XML karena SOAP umumnya menggunakan protokol HTTP sebagai sarana *transport* datanya dan data yang akan dipertukarkan ditulis dalam format XML. Karena SOAP menggunakan HTTP dan XML maka SOAP memungkinkan pihak-pihak yang mempunyai *platform*, sistem operasi dan perangkat lunak yang berbeda dapat saling mempertukarkan datanya. SOAP mengatur bagaimana *request* dan *respons* dari suatu *web service* bekerja [5].

Peran SOAP adalah sebagai protokol yang dimaksudkan untuk bertukar pesan yang berbasis XML antara aplikasi dengan desentralisasi dan terdistribusi [6]. Spesifikasi yang digunakan tidak lebih seperti sebuah amplop biasa berbasis XML untuk informasi yang ditransfer, serta sekumpulan aturan bagi translasi aplikasi dan tipe-tipe data *platform* yang spesifik menjadi bentuk XML. Desain bentuk dari SOAP membuatnya cocok untuk berbagai pertukaran pesan pada aplikasi. SOAP dan beberapa teknologi yang didukung seperti WSDL (*Web Service Description Language*) dan UDDI (*Universal Description Discovery and Integration*) merupakan kombinasi dari XML yang dikirimkan melalui HTTP [7].

Sebuah SOAP adalah sebuah dokumen XML yang terdiri dari Tiga elemen utama [8] seperti pada Gambar 2.

Gambar 2. Elemen utama SOAP [8]

Seperti yang tersaji pada Gambar 2 masing-masing komponen tersebut, yaitu:

1. *Envelope element*, yaitu elemen yang mengidentifikasi dokumen XML pada

pesan SOAP yang bertugas membungkus isi pesan SOAP.

2. *Element header*, yaitu elemen yang bersifat opsional berisi informasi yang bukan merupakan bagian dari pesan itu sendiri.
3. *Element body*, berisi permintaan aktual dan merespon informasi.

```
<?xml version="1.0" ?>
<env:Envelope xmlns:env="http://www.w3.org/2001/12/soap-envelope">
  <env:Header>
    <n:alertcontrol xmlns:n="http://example.org/alertcontrol">
      <n:priority>1</n:priority>
      <n:expires>2001-06-22T14:00:00-05:00</n:expires>
    </n:alertcontrol>
  </env:Header>
  <env:Body>
    <m:alert xmlns:m="http://example.org/alert">
      <m:msg>Pick up Mary at school at 2pm</m:msg>
    </m:alert>
  </env:Body>
</env:Envelope>
```

2.4. WSDL (*Web Service Definition Language*)

WSDL adalah *grammar* XML untuk mendeskripsikan antar muka *web service*, dengan protokol yang mendukung layanan lokasi *web service*. WSDL akan memfasilitasi komunikasi antar aplikasi. WSDL akan mendeskripsikan apa yang akan dilakukan oleh *web service*, bagaimana menemukannya dan bagaimana untuk mengoperasikannya. Spesifikasi WSDL versi 1.1 dibuat oleh IBM Research dan Microsoft [9].

Spesifikasi WSDL mendefinisikan tujuh tipe elemen yaitu *type*, *message*, *operation*, *port type*, *binding*, *port* dan *service* [10].

Adapun komponen WSDL antara lain sebagai berikut:

1. *Interface* mendefinisikan antarmuka dari *web service* dan terdiri dari beberapa operasi yang dimiliki.
2. *Operations* merepresentasikan fungsi dan operasi yang dimiliki oleh *web service* dan dapat mencatu pada banyak komponen *message*.

3. *Messages* mendefinisikan koleksi dari parameter masukan dan keluaran dan dapat mengandung banyak komponen *parts*.
4. *Parts* merepresentasikan data masukan dan keluaran yang digunakan komponen *operations*.
5. *Services* merupakan kumpulan dari *endpoint* yang didefinisikan dalam masing-masing komponen *endpoint*.
6. *Endpoint* mengandung data dari *endpoint*, termasuk alamat fisik dan informasi protokol. *Binding* menghubungkan enam komponen di atas dengan komponen *operation*.

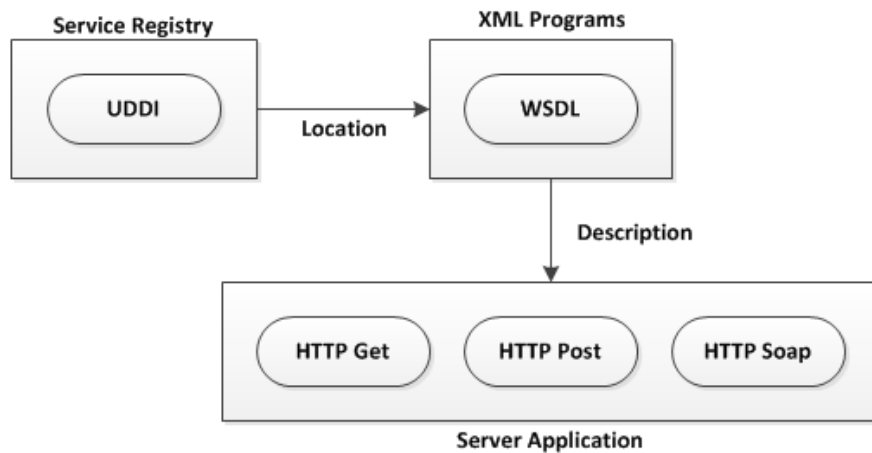
2.5. UDDI (*Universal Description Discovery and Integration*)

UDDI merupakan *registry* untuk menyimpan informasi tentang *web service*. *XML-based registry* berisi nama-nama dari organisasi tersebut, beserta *service* dan deskripsi dari *service* yang mereka berikan. UDDI berbasis XML, berjalan pada HTTP dan mengadopsi pesan SOAP untuk pemrograman *cross-platform* dan

menggunakan WSDL untuk mendeskripsikan layanan *web* [11].

UDDI mengkombinasikan SOAP dan WSDL untuk pembentukan sebuah *registry* API bagi pendaftaran dan pengenalan *web service*. UDDI

menyediakan sebuah area umum di mana sebuah organisasi dapat mengiklankan keberadaan mereka dan *service* yang diberikan [11]. Gambar 3 menunjukkan skema proses kombinasi SOAP dan WSDL untuk sebuah *registry*.



Gambar 3. Kombinasi UDDI, SOAP dan WSDL untuk pembentukan *registry* [3]

2.6. Rumusan Masalah

Berdasarkan latar belakang masalah di atas, maka dapat dirumuskan masalah utama yang dihadapi adalah bagaimana merancang dan membuat suatu sistem informasi dengan basis data yang terintegrasi dengan data yang aktual, antara puskesmas dan Dinas Kesehatan Kabupaten Cirebon untuk meningkatkan kinerja Dinas Kesehatan dalam pelaporan bulanan data penyakit.

2.7. Analisa dan Rancangan

Suatu sistem dibangun berdasarkan kebutuhan informasi dalam suatu organisasi, oleh sebab itu pada umumnya perancangan sistem dimulai dari pengamatan kebutuhan informasi. Sistem yang dirancang berfungsi dalam menangani integrasi data laporan bulanan penyakit dari puskesmas-puskesmas ke Dinas Kesehatan. Sistem integrasi untuk mengolah data laporan bulanan penyakit dari puskesmas-puskesmas terbagi menjadi Dua yaitu:

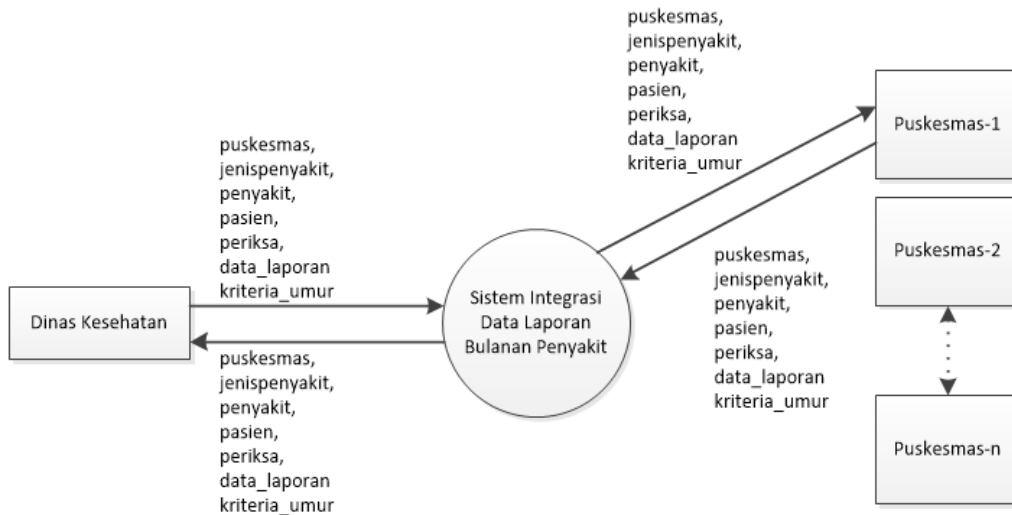
1. Sistem integrasi data laporan bulanan penyakit pada puskesmas-puskesmas

sebagai *client* yang meminta layanan ke Dinas Kesehatan untuk melakukan replikasi data.

2. Sistem integrasi data pada Dinas Kesehatan berfungsi sebagai *server* yang menyediakan metode *web service* untuk memberikan layanan replikasi data.

Gambar 4. menampilkan arsitektur sistem yang disimulasikan ke dalam 3 jenis aplikasi *server*. Data integrasi bersumber dari data masing-masing puskesmas mewakili *web server* di tiap puskesmas. Setiap aplikasi ini terhubung pada *database* yang akan berperan untuk pengiriman data ke Dinas Kesehatan Kabupaten Cirebon.

Berdasarkan pada Gambar 4, sistem dirancang dengan mekanisme pengiriman data dari *server* puskesmas-puskesmas ke *server* Dinas Kesehatan Kabupaten Cirebon. Integrasi data dilakukan dengan teknik replikasi data berbentuk sumber *database* di Dinas Kesehatan Kabupaten Cirebon.



Gambar 5. Diagram konteks (DFD level 0)

3. HASIL DAN PEMBAHASAN

Komunikasi antara puskesmas-puskesmas yang bertindak sebagai *client* dan Dinas Kesehatan yang bertindak sebagai *server* dilakukan melalui replikasi *database*. Proses komunikasi yang terjadi adalah *client* menyimpan data pada basis data lokal menggunakan MySQL, dan apabila data telah disimpan secara lokal pada puskesmas, maka dapat dilakukan sinkronisasi melalui replikasi basis data di Dinas Kesehatan Kabupaten Cirebon menggunakan MySQL. Proses penyimpanan data dan sinkronisasi ini dilakukan secara langsung ke *server* melalui pemanfaatan *web service*. Hasil *database* di Dinas Kesehatan memiliki bentuk yang homogen dengan data yang ada di puskesmas.

Implementasi integrasi data menggunakan replikasi data dilakukan setelah transaksi data berhasil dikerjakan pada masing-masing puskesmas. Ketika transaksi data dikerjakan di puskesmas yang bertindak sebagai *client* telah berhasil

melakukan replikasi data, maka *client* akan melakukan *request* ke *server* Dinas Kesehatan untuk me-*response* permintaan replikasi data. Berikut ini disajikan beberapa transaksi data di puskesmas-puskesmas.

3.1. Sistem Integrasi Data

Sistem yang dihasilkan pada penelitian merupakan integrasi data yang bertujuan membantu laporan bulanan penyakit pada puskesmas ke Dinas Kesehatan yang dibuat untuk bekerja pada akses *internet*. Halaman *interface* sistem integrasi data dirancang dapat diakses oleh *user* yang mempunyai hak akses atau antar muka halaman utama.

3.2. Halaman data puskesmas

Pada sistem integrasi data yang ada di Dinas Kesehatan *user* dapat menambah, mengubah dan menghapus data puskesmas yang ada dalam *database*. Halaman menu objek data puskesmas pada Dinas Kesehatan disajikan pada Gambar 6.

Data Puskesmas					
<input type="button" value="Tambah"/>					
No.	ID Puskesmas	Nama Puskesmas	Kelurahan	Kecamatan	Operasi
1	P320916202	ASTAPADA	ASTAPADA	TENGAH TANI	
2	P320916201	KEDAWUNG	KEDAWUNG	KEDAWUNG	

Gambar 6. Halaman menu objek puskesmas pada Dinas Kesehatan

3.3. Halaman data penyakit

Halaman ini ditampilkan setelah *user* memilih sub menu Jenis Penyakit dari menu Data Master. Sistem yang disajikan tersebut menampilkan semua jenis penyakit yang ada pada *database*. Pada sistem

integrasi data yang ada di puskesmas *user* dapat menambah, mengedit, menghapus serta mengambil/mengimpor data jenis penyakit dari *database* yang ada di *server* Dinas Kesehatan. Sistem layanan pada puskesmas ini disajikan pada Gambar 7.

No.	ID Penyakit	Nama Penyakit	ID Jenis	Nama Jenis Penyakit	Operasi
1	A00	Kolera	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
2	A01.0	Demam Tifoid	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
3	A01.4	Demam Paratifoid, Tidak spesifik	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
4	A02	Infeksi Salmonela lainnya	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
5	A03	Shigelosis; Disentri Basiler tidak spesifik	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
6	A04	Infeksi Usus karena bakteri lainnya tidak spesifik	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
7	A05	Keracunan Makanan karena bakteri lainnya	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
8	A06	Amubiasis, Disentri Amuba	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	

Gambar 7. Halaman menu objek penyakit di puskesmas

Sistem integrasi data yang ada di Dinas Kesehatan menyediakan *service*, di mana puskesmas bisa mengimpor data jenis

penyakit untuk disimpan di *database* puskesmas. Sistem layanan pada Dinas Kesehatan ini disajikan pada Gambar 8.

No.	ID Penyakit	Nama Penyakit	ID Jenis	Nama Jenis Penyakit	Operasi
1	A00	Kolera	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
2	A01.0	Demam Tifoid	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
3	A01.4	Demam Paratifoid, Tidak spesifik	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
4	A02	Infeksi Salmonela lainnya	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
5	A03	Shigelosis; Disentri Basiler tidak spesifik	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
6	A04	Infeksi Usus karena bakteri lainnya tidak spesifik	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
7	A05	Keracunan Makanan karena bakteri lainnya	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	
8	A06	Amubiasis, Disentri Amuba	01	PENYAKIT INFEKSI DAN PARASIT TERTENTU	

Gambar 8. Halaman menu objek penyakit di dinas kesehatan

3.4. Halaman data laporan

Halaman ini ditampilkan setelah *user* memilih menu Laporan. Sistem yang disajikan tersebut untuk membuat laporan baru dan menampilkan data laporan yang sudah tersimpan sebelumnya. Laporan bulanan yang dihasilkan merupakan data-

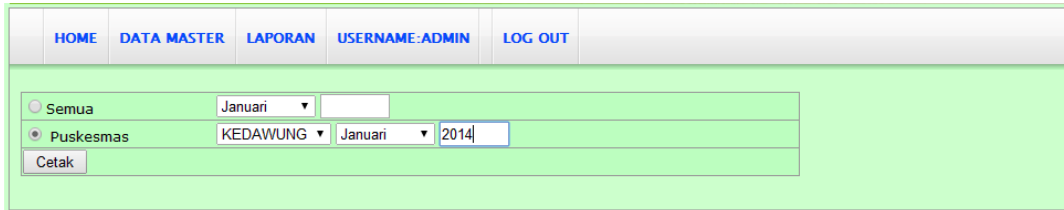
data yang diperoleh dari proses pemeriksaan pasien dan disimpan dalam *database*. Laporan yang dibuat tersebut selanjutnya dikirimkan ke *server* Dinas Kesehatan. Sistem layanan ini disajikan pada Gambar 9.

Aksi	ID	Penyakit	Id	0-7hr	8-28	29-11th	1-4th	5-9th	10-14th	15-19th	20-44th	45-54th	55-59th
	A20	Pes/Sampar	4	0	0	0	1	0	0	0	0	0	0

Gambar 9. Halaman menu objek laporan di puskesmas

Setelah laporan dari puskesmas dikirimkan ke *server* Dinas Kesehatan maka laporan tersebut ditampilkan dari

server Dinas Kesehatan dengan memilih menu Laporan seperti disajikan pada Gambar 10.



Gambar 10. Halaman menampilkan laporan di dinas kesehatan

Berdasarkan pengujian yang dilakukan, waktu yang diperlukan untuk menampilkan laporan pada sistem yang ada di dinas kesehatan cukup singkat yaitu rata-rata 10 detik.

Dokumen WSDL menyediakan *metadata* untuk suatu *service*. NuSOAP mengizinkan *user* untuk menentukan WSDL yang digenerate untuk suatu layanan dengan menggunakan beberapa

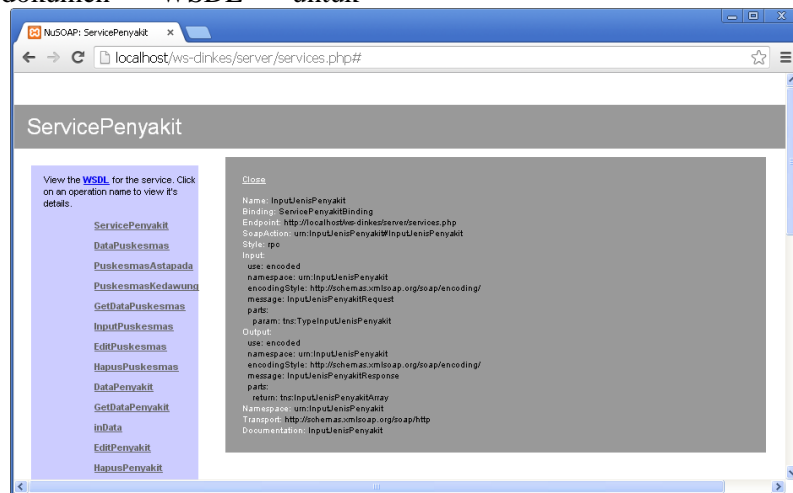
method dan *class soap_server*. WSDL dideklarasikan pada beberapa bagian utama, untuk menghasilkan informasi pada *method* yang tersedia pada layanan. *Method* yang tersedia pada layanan, dispesifikasikan dengan parameter tambahan pada fungsi *register*. Listing 2 adalah proses generate WSDL dengan mengeksekusi *method* “*configure WSDL*”

Listing 2. Potongan program generate WSDL pada servicePenyakit

```
$server = new soap_server();
$server->configureWSDL('ServicePenyakit', 'urn:ServicePenyakit');
$server->wsdl->schemaTargetNamespace = 'urn:ServicePenyakit';
```

Nilai-nilai yang dikembalikan dari layanan di atas berbentuk *array*, sehingga tipe data yang digunakan juga *array*. Jika file *response web service* dijalankan, maka informasi dokumen WSDL untuk

“*InputJenisPenyakit*” dapat dilihat pada Gambar 11. Detail spesifikasi dari layanan dapat diketahui dengan mengklik link “*InputJenisPenyakit*”.



Gambar 11. Informasi dokumen WSDL dengan *method* “*InputJenisPenyakit*”

3.5. Method Web Service pada Layanan Integrasi Data

Teknologi *web service* memungkinkan kita dapat menghubungkan berbagai jenis *software* yang memiliki platform dan sistem operasi yang berbeda. Jadi, kita tetap mendapatkan sebuah potongan informasi dari suatu website tanpa harus mengunjungi website tersebut, cukup dengan mengetahui fungsi/*method web service* yang disediakan oleh *website* tersebut. *User* dapat mengakses potongan

informasi itu dengan meletakkan fungsi/*method* pada aplikasi.

Method yang tersedia untuk mengakses objek dan tipe objek. Dokumen SOAP *server* dan NuSOAP akan menghasilkan layanan *web method* yang menghasilkan data *array* sesuai parameternya. Sistem integrasi data laporan bulanan penyakit ini mempunyai beberapa *method web service* utama yang ditunjukkan pada Tabel 1.

Tabel 1. *Method web service* data integrasi

No	<i>Method</i>	Deskripsi
1	InputPuskesmas	Penambahan <i>record</i> data puskesmas
2	EditPuskesmas	Perubahan <i>record</i> data puskesmas
3	HapusPuskesmas	Penghapusan <i>record</i> data puskesmas
4	InputJenisPenyakit	Penambahan <i>record</i> data jenis penyakit
5	EditJenisPenyakit	Perubahan <i>record</i> data jenis penyakit
6	HapusJenisPenyakit	Penghapusan <i>record</i> data jenis penyakit
7	InputPenyakit	Penambahan <i>record</i> data penyakit
8	EditPenyakit	Perubahan <i>record</i> data penyakit
9	HapusPenyakit	Penghapusan <i>record</i> data penyakit
10	InputLaporan	Penambahan <i>record</i> data laporan
11	EditLaporan	Perubahan <i>record</i> data laporan
12	HapusLaporan	Penghapusan <i>record</i> data laporan

3.6. Pengujian proses sistem integrasi

Komunikasi antara *client* dan *server* dilakukan melalui replikasi database. Puskesmas-puskesmas berperan sebagai *client* yang meminta respon *server* dari *web service*. Adapun peran Dinas Kesehatan Kabupaten Cirebon sebagai *server* yang menggunakan teknologi *web service* dengan PHP.

Tabel 2 menampilkan pengujian yang telah dilakukan serta hasilnya telah sesuai dengan apa yang diharapkan. Percobaan Pertama yang mewakili objek data dilakukan pada objek data jenis penyakit. Hasil pengujian replikasi data jenis penyakit dari server Dinas Kesehatan ke puskesmas dengan jumlah record 20 memerlukan waktu 4 detik.

Tabel 2. Pengujian pada objek data jenis penyakit

Input Pengujian	Hasil yang diharapkan	Status
Melakukan invokasi pada <i>web method</i> "InputJenisPenyakit" pada salah satu <i>client</i> dengan input <i>insert</i> ke <i>database</i> .	Tabel jenis penyakit pada basis data <i>server</i> puskesmas mengalami penambahan <i>record</i> sesuai <i>query input</i> .	Terpenuhi
Melakukan invokasi pada <i>web method</i> "EditJenisPenyakit" pada salah satu <i>client</i> dengan input <i>edit</i> ke <i>database</i> .	Tabel jenis penyakit pada basis data <i>server</i> puskesmas mengalami perubahan <i>record</i> sesuai <i>query edit</i> .	Terpenuhi
Melakukan invokasi pada <i>web</i>	Tabel jenis penyakit pada basis	Terpenuhi

<i>method</i> “HapusJenisPenyakit” pada salah satu <i>client</i> dengan input <i>delete</i> ke <i>database</i> .	data <i>server</i> puskesmas mengalami penghapusan <i>record</i> sesuai <i>query delete</i> .	
--	---	--

Tabel 3 adalah tabel yang menampilkan pengujian yang telah dilakukan serta hasilnya telah sesuai dengan apa yang diharapkan. Percobaan Kedua dilakukan dengan invokasi pada objek data

penyakit. Hasil pengujian replikasi data penyakit dari server Dinas Kesehatan ke puskesmas dengan jumlah record 220 memerlukan waktu 41 detik.

Tabel 3. Pengujian pada objek data penyakit

Input Pengujian	Hasil yang diharapkan	Status
Melakukan invokasi pada <i>web method</i> “InputPenyakit” pada salah satu <i>client</i> dengan input <i>insert</i> ke <i>database</i> .	Tabel penyakit pada basis data <i>server</i> puskesmas mengalami penambahan <i>record</i> sesuai <i>query</i> input.	Terpenuhi
Melakukan invokasi pada <i>web method</i> “EditPenyakit” pada salah satu <i>client</i> dengan input <i>edit</i> ke <i>database</i> .	Tabel penyakit pada basis data <i>server</i> puskesmas mengalami perubahan <i>record</i> sesuai <i>query edit</i> .	Terpenuhi
Melakukan invokasi pada <i>web method</i> “HapusPenyakit” pada salah satu <i>client</i> dengan input <i>delete</i> ke <i>database</i> .	Tabel penyakit pada basis data <i>server</i> puskesmas mengalami penghapusan <i>record</i> sesuai <i>query delete</i> .	Terpenuhi

Tabel 4 adalah tabel yang menampilkan pengujian yang telah dilakukan serta hasilnya telah sesuai dengan apa yang diharapkan. Percobaan Ketiga dilakukan dengan invokasi pada objek data

laporan. Hasil pengujian replikasi data laporan bulanan penyakit dari *server* puskesmas-puskesmas ke Dinas Kesehatan dengan jumlah data 35 record memerlukan waktu 21 detik.

Tabel 4. Pengujian pada objek data laporan

Input Pengujian	Hasil yang diharapkan	Status
Melakukan invokasi pada <i>web method</i> “InputLaporan” pada salah satu <i>client</i> dengan input <i>insert</i> ke <i>database</i> .	Tabel laporan pada basis data <i>server</i> Dinas Kesehatan mengalami penambahan <i>record</i> sesuai <i>query</i> input.	Terpenuhi
Melakukan invokasi pada <i>web method</i> “EditLaporan” pada salah satu <i>client</i> dengan input <i>edit</i> ke <i>database</i> .	Tabel laporan pada basis data <i>server</i> Dinas Kesehatan mengalami perubahan <i>record</i> sesuai <i>query edit</i> .	Terpenuhi
Melakukan invokasi pada <i>web method</i> “HapusLaporan” pada salah satu <i>client</i> dengan input <i>delete</i> ke <i>database</i> .	Tabel laporan pada basis data <i>server</i> Dinas Kesehatan mengalami penghapusan <i>record</i> sesuai <i>query delete</i> .	Terpenuhi

Berdasarkan skenario dan hasil pengujian yang telah dilakukan terlihat bahwa mekanisme pengekseskuan *method* dapat dilakukan dengan benar. Pengujian ini telah membuktikan bahwa validitas dari

masing-masing unit proses di *client* dapat membentuk keseluruhan sistem menjadi terintegrasi dalam bentuk replikasi *database* di *server*.

4. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, dapat diperoleh beberapa kesimpulan:

1. Sistem integrasi data yang dikembangkan telah mengintegrasikan aplikasi layanan dari sistem berbeda pada puskesmas-puskesmas dan Dinas Kesehatan dengan fungsi-fungsi *web service* dan meningkatkan kinerja laporan bulanan data penyakit.
2. Teknologi *web service* pada sistem integrasi dengan teknik replikasi data telah berhasil melakukan pertukaran data antar *server* puskesmas dan Dinas Kesehatan dalam menyajikan laporan bulanan penyakit.
3. Sistem integrasi data dengan metode *web service* telah menunjukkan kinerja yang cukup baik karena bisa mereplikasi data antar server, yaitu data jenis penyakit 20 *record* dalam waktu 4,14 detik, data penyakit 220 *record* dalam waktu 41 detik dan data laporan bulanan penyakit 35 *record* dalam waktu 21 detik.

5. SARAN

Beberapa saran yang penulis sampaikan apabila sistem ini akan dikembangkan lebih lanjut:

1. Sistem keamanan *web service* dapat dikembangkan lebih lanjut untuk menjamin kerahasiaan data dan informasi yang mengalir di jaringan *internet*.
2. Perlu adanya standar dalam penyusunan laporan data bulanan penyakit untuk memudahkan integrasi data dan kualitas jaringan yang baik dan pemeliharaan secara kontinyu baik sisi *software* maupun *hardware*.

DAFTAR PUSTAKA

- [1] Gottschalk, K., Graha, S., Kreger, H., Snell, J., 2002, *Introduction to Web Service Architecture*, IBM System Journal, IBM, 41, 2.
- [2] Cheng, Y.H., 2008, *Research of Solving Scheme of E-Commerce System Based on Web Services*, IEEE Computer Society, 230-234.
- [3] Chen, M., 2003, *Factors Affecting the Adoption and Diffusion of XML, and Web Services Standard for E-Business Systems*, Int. J. Human-Computer Studies, 58, 257-279.
- [4] Junedi, M., 2003, *Pengantar XML*, Kuliah Umum IlmuKomputer.com
- [5] Subramoney, K.P., 2009, *A Secure Client/Server Interface Protocol for the Electricity Prepayment Vending Industry*, Dissertation, University of Pretoria, South Africa.
- [6] Kim, L., 2002, *XML Integrated Development Environment*, Altova, The XML Spy Company, USA.
- [7] Khalili, A., Badrabi A.H. dan Khoshalhan, F. 2008, *A Framework for Distributed Market Place Based on Intelligent Software Agents and Semantics Web Services*, IEEE Congress on Services Part II, 141-149.
- [8] Poollmann, C.G. dan Claessens, J., 2005, *Web Services and Web Services Security Standards*, Information Security Technical Report, 10, 15-24.
- [9] Akkiraju, R., Farrel, J., Miller, J., Nagarajan, M., Schmidt, M.T., Sheth, A. dan Verma, K., 2005, *Web Service Semantics-WSDLs-S*, A Joint UGA-IBM Technicl Note, IBM, Georgia.
- [10] Vara, J.M., Castro, V.D. dan Marcos, E., 2005, *WSDL Automatic Generation from UML Models in a MDA Framework*, International Journal of Web Services Practices, 1, 1-12.
- [11] Bouzakis, K.D., Andreadis, G., Vakali, A. dan Sarigiannidou, M., 2007, *Automating the Manufacturing Process Under a Web Based Framework*, Advances In Engineering Software, 40, 956-964.