

INTEGRASI ANTAR SISTEM INFORMASI YANG HETEROGEN MENGUNAKAN METODE WEB SERVICE SOAP

MUHAMAD FATKHUR ROHIM

Program Studi Teknik Informatika - S1, Fakultas Ilmu Komputer, Universitas Dian

Nuswantoro Semarang

Abstrak

Computer-based information systems has become a fundamental requirement in a particular agency or Organization, As the rapid development of technology to produce new products or there is a change from previous technologies, for example, a database that is used in many kinds at the information systems. But every databases has a different specification. The design of the information system is dedicated according with the organization itself, but this time as a result of changes in existing business process model requires a mutual information system integration with other information systems. To perform the integration is not possible to change or rebuild existing information Systems. That's the basis of the research, the integration of data between heterogeneous information systems on the type of Database Management System (DBMS), naming tables and structure attributes of different tables, the authors use the web service technique is one method that can be used to integrate information between applications or systems with each other without knowing how the system architecture will be integrated. Web service operability has properties for use standard XML data format in communication, the HTTP transport protocol and to handle data communication using the SOAP protocol.

Keywords: Integration of data, heterogeneity of Information Systems, Database Management System (DBMS), Web Services, SOAP, XML

Key word : Integration of data, heterogeneity of Information Systems, Database Management System (DBMS), Web Services, SOAP, XML

1. Pendahuluan

Pada era modern ini perkembangan teknologi komunikasi dan informasi sudah semakin pesat dan maju. Teknologi ini digunakan untuk mempermudah pekerjaan manusia dengan bantuan komputer. Perkembangan teknologi komputer meliputi dari perangkat keras dan perangkat lunak. Ada banyak penemuan teknologi baru, perbaikan dari teknologi versi sebelumnya bahkan ada perubahan mendasar dari teknologi yang sudah ada, sehingga menghasilkan jenis *platform* baru yang masing-masing memiliki keunggulan. Dalam perkembangan perangkat lunak banyak *platform* basis data baru, baik yang berbayar maupun gratis yang memiliki spesifikasi berbeda, jenis basis data ini yang akan diterapkan pada sebuah sistem informasi.

Sekarang ini dalam sebuah organisasi sudah memanfaatkan teknologi sistem informasi karena informasi menjadi kepentingan mendasar pada kehidupan manusia. Informasi didapat dari hasil komunikasi antar beberapa individu atau kelompok secara langsung. Namun setelah adanya teknologi internet dan digabungkan dengan sebuah sistem informasi berbasis komputer, komunikasi bisa dilakukan secara jarak jauh dengan waktu yang bersamaan, sehingga memudahkan dalam mendapatkan informasi.

Banyak sistem informasi yang telah dirancang dan dibangun di beberapa organisasi, instansi, perusahaan, dan dunia akademik dengan menggunakan perancangan yang sesuai dengan kebutuhannya masing-masing. Namun seiring dengan kebutuhan proses model bisnis yang terjadi pada

organisasi yang menuntut sistem informasi suatu organisasi saling integrasi dengan sistem informasi yang lain. [1] Padahal masing-masing sistem yang dibangun memiliki perbedaan arsitektur, jenis basis data yang digunakan, penamaan tabel pada basis data, serta struktur atribut tabel yang berbeda. Serta tidak ada hak akses sistem lain untuk bisa mengakses secara langsung dengan melakukan *query* ke basis data lain karena privasi dan keamanan. [1] Sehingga muncul hambatan untuk melakukan integrasi.

Saat ini telah berkembang sebuah teknologi yang memungkinkan komunikasi antar satu aplikasi dengan aplikasi lainnya. Yaitu suatu program atau aplikasi komputer berupa fungsi yang melakukan sebuah tugas secara spesifik, kemudian fungsi atau *service* ini di bagikan ke aplikasi lain dengan proses transfer data menggunakan XML yang dikenal dengan *web service* [2]. Maka teknologi *web service* ini digunakan untuk mengeliminir masalah pada sistem informasi yang memiliki perbedaan pada jenis basis data, nama tabel, serta struktur atribut basis data bisa saling terhubung karena tidak mungkin menghapus sistem informasi yang sudah dibangun kemudian membangun ulang suatu sistem dengan menyamakan jenis basis data, nama tabel data dan struktur tabel pada basis data. [1]

Arsitektur yang ada pada *web service* salah satunya adalah *Simple Oriented Architecture* (SOA), yang memungkinkan kita untuk dapat terhubung berbagai jenis perangkat lunak yang memiliki basis data dan sistem operasi yang berbeda untuk saling bertukar data [1]. Hal ini memungkinkan karena *web service* yang menggunakan architecture SOA menggunakan sebuah standar format data yang universal yaitu XML *web service* [3]

Dari uraian di atas, maka penulis ingin melakukan sebuah perancangan dan pembangunan sistem integrasi antar sistem informasi yang memiliki jenis basis data yang berbeda, nama tabel yang berbeda serta

struktur tabel yang berbeda dengan cara membuat suatu *service* pada sistem untuk melakukan tugas secara spesifik kemudian *service* tadi dibagi ke sistem lain dengan simulasi pembuatan beberapa sistem berbeda. Model arsitektur *web service* dengan menggunakan *Simple Oriented Architecture* (SOA) dan protokol web service yang digunakan adalah *Simple Object Access Protocol* (SOAP).

Dari beberapa sistem informasi yang sudah dibangun yang memiliki perbedaan *platform* jenis DBMS, nama tabel, serta struktur atribut tabel. Bagaimana merancang dan membangun sistem integrasi yang heterogen yang telah di uraikan tadi, supaya bisa saling integrasi dengan menggunakan metode *web service* SOAP?

Adapun batasan masalah pada penelitian ini adalah sebagai berikut:

1. Pembuatan web service tidak sampai keamanan enkripsi.
2. Heterogenitas dari sistem yang akan diintegrasikan pada platform jenis Database Manipulation Language (DBMS), perbedaan nama tabel pada basis data, atau perbedaan struktur atribut tabel.
3. Dibuat prototype 4 aplikasi yang memiliki platform berbeda yaitu pada jenis Database Manipulation Language (DBMS), perbedaan nama tabel pada basis data, perbedaan struktur atribut tabel. Kemudian dihubungkan dengan web service untuk simulasi dan membuktikan inter-operability.

2. Tinjauan Pustaka

2.1 Integrasi Data

Integrasi aplikasi *enterprise* merupakan pendekatan strategis untuk membungkus beberapa sistem informasi bersama-sama dan mendukung kemampuan untuk bertukar informasi secara *real time*. Integrasi aplikasi ini adalah proses mengintegrasikan berbagai aplikasi independen, tujuannya adalah untuk berbagi data dan proses dapat dilakukan

tanpa melakukan perubahan terhadap aplikasi-aplikasi dan struktur datanya, yang mana hal ini merupakan efektifitas dalam hal pembiayaan [5].

2.2 Sistem Homogen dan Heterogen

Sistem homogen artinya sistem database terdistribusi yang menghubungkan DBMS yang bertipe sama, kompatibel, struktur dan deskripsi datanya sama. Sedangkan yang menghubungkan DBMS yang tidak kompatibel adalah sistem yang heterogen. Sedangkan sistem heterogen membutuhkan standar yang dapat menginterkoneksi database dari pembuat yang berbeda. [10]

2.3 Web Service

Web service adalah sekumpulan fungsionalitas operasi yang dapat diakses dalam sebuah jaringan melalui pesan XML yang telah distandarkan. W3C mendefinisikan web service sebagai sebuah sistem perangkat lunak yang dirancang untuk mendukung inter-operability mesin ke mesin di sebuah jaringan. Web service merupakan komponen perangkat lunak loosely coupled, dapat digunakan ulang, kemudian didistribusikan, dan dapat diakses dengan menggunakan protokol internet yang telah distandarkan. Web service biasanya digunakan untuk eliminir pada suatu sistem karena ada perbedaan platform yang digunakan agar masing-masing sistem saling integrasi [2].

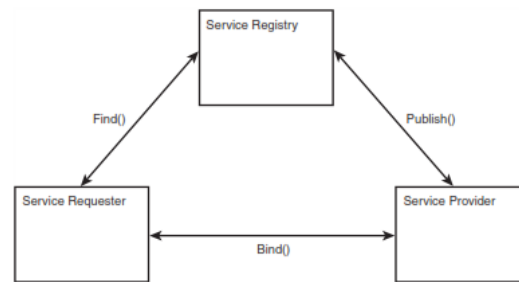
2.4 XML

Extensible Markup Language (XML) merupakan bahasa *markup*, yang menggunakan *tag* sebagai tanda, kategori, dan mengorganisasi informasi secara spesifik. *Markup* mendeskripsikan dokumen atau struktur data dan organisasi. Content, seperti teks, gambar, dan data merupakan dari kode yang mengandung banyak *markup tag*. XML tidak terbatas pada kumpulan *markup* tertentu. Fleksibilitas XML membawa pada tersebar luasnya pemakaian pertukaran data pada banyak *form* [12].

2.5 SOA

The Service Oriented Architecture (SOA) memberikan model teori untuk semua *web service*. SOA memiliki model yang simpel,

berisi tiga entitas dan tiga operasi. Seperti terlihat pada gambar di bawah ini [1].



Gambar 1: Model Arsitektur SOA

2.6 SOAP

Simple Object Access Protocol (SOAP) adalah protokol untuk saling bertukar pesan dalam format XML antar komputer di dalam jaringan, biasanya menggunakan HTTP/HTTPS. Dalam penelitian ini menggunakan SOAP sebagai protokol pengiriman pesannya. Dalam menjalankan tugasnya, SOAP menggunakan struktur XML tertentu dalam mengirimkan paket request kepada web service. Penerimaan respon dari web service pun memiliki strukturnya sendiri [1].

3. Analisis dan Perancangan

3.1 Analisis Sistem Integrasi

Analisis dari sistem integrasi dari beberapa sub sistem yang heterogen adalah sebagai berikut:

- Integrasi antara Sistem Kependudukan dengan Sistem Puskesmas.
- Integrasi antara Sistem Kependudukan dan Sistem Rumah Sakit.
- Integrasi antara Sistem Puskesmas dengan Sistem Dinas Kesehatan.
- Integrasi antara Sistem Rumah Sakit dengan Sistem Dinas Kesehatan.

3.2 Identifikasi Masalah

Ada permasalahan yang timbul dari interaksi antar sistem tadi yaitu sebagai berikut:

- Masalah integrasi antara sistem Dinas Kependudukan dan sistem Puskesmas dalam menggunakan data kependudukan yang akan digunakan menjadi data pasien di Puskesmas. Ada

beberapa masalah yang timbul yaitu sebagai berikut:

1. Untuk mengambil data penduduk tidak mungkin melakukan query langsung ke database sistem Kependudukan karena alasan privasi.
 2. Perbedaan nama tabel. Nama tabel pada sistem Kependudukan adalah tabel_penduduk, sedangkan tabel pasien yang ada pada sistem Puskesmas adalah t_pasien.
 3. Perbedaan struktur atribut tabel antara tabel_penduduk dengan t_pasien pada sistem Puskesmas.
- b. Masalah integrasi antara sistem Rumah sakit dengan sistem Kependudukan dalam menggunakan data penduduk yang akan digunakan untuk membuat data pasien di sistem Rumah sakit, ada beberapa masalah yaitu sebagai berikut:
1. Untuk mengambil data penduduk tidak mungkin sistem Rumah sakit langsung melakukan query ke sistem Kependudukan, karena alasan privasi data.
 2. Perbedaan jenis Database Management System (DBMS), pada sistem Kependudukan DBMS yang digunakan adalah MySQL sedangkan sistem Rumah sakit menggunakan DBMS PostgreSQL.
 3. Perbedaan nama tabel, nama tabel pada sistem Kependudukan yaitu tabel_penduduk, sedangkan nama tabel pada sistem Rumah sakit adalah tabel_pasien.
 4. Perbedaan struktur atribut tabel antara tabel_penduduk dan tabel_pasien. Berikut adalah struktur atribut tabel penduduk pada sistem Kependudukan.
- c. Masalah integrasi dan sinkronisasi laporan dari Puskesmas ke Dinas Kesehatan. Ada beberapa masalah yaitu sebagai berikut:
1. Untuk mengambil data – data laporan tidak memungkinkan untuk melakukan query langsung ke database sistem Puskesmas.
- d. Masalah integrasi dan sinkronisasi antar sistem Rumah sakit dengan sistem Dinas Kesehatan adalah sebagai berikut :
1. Untuk mengambil data – data laporan tidak memungkinkan untuk melakukan query langsung ke database sistem Puskesmas.
 2. Ada perbedaan jenis Database Management System (DBMS), pada sistem Rumah sakit menggunakan DBMS PostgreSQL sedangkan pada sistem Dinas Kesehatan menggunakan DBMS Mysql.
 3. Ada perbedaan pada arsitektur sistem, penamaan tabel yang berbeda, serta struktur tabel yang berbeda.
- e. Masalah integrasi antar Sistem Dinas Kesehatan untuk memerikan data ICD penyakit yang telah distandarkan adalah sebagai berikut:
1. Untuk mengambil data – data laporan tidak memungkinkan untuk melakukan query langsung ke database sistem Dinas Kesehatan.
 2. Adanya perbedaan penggunaan Database Management System dari sistem yang akan mengambil data.
 3. Perbedaan penamaan tabel, serta struktur atribut tabel yang berbeda.

3.3 Usulan Pemecahan Masalah

Ada beberapa usulan untuk memberikan solusi dari permasalahan di atas, diantara yaitu sebagai berikut:

- a. Integrasi data penduduk antara sistem Kependudukan dengan Sistem Puskesmas dan Sistem Rumah Sakit dibuat *service* untuk membagi data penduduk ke sistem lain menggunakan *web service* SOAP dengan komunikasi data menggunakan XML. Berikut adalah web service yang diberikan oleh sistem Kependudukan.

1. getPenduduk

- b. Integrasi dan sinkronisasi antara sistem Puskesmas dengan sistem Dinas Kesehatan dalam proses monitoring data laporan menggunakan *service* data laporan dan *service* data penderita menggunakan web service SOAP dengan komunikasi data XML. Berikut adalah web service sistem Puskesmas untuk melihat data laporan penyakit dan laporan penderita penyakit.

1. getLaporanW1
2. getLaporanW2
3. getLaporanLB1
4. getPenderitaW1
5. getPenderitaW2
6. getPenderitaLB1

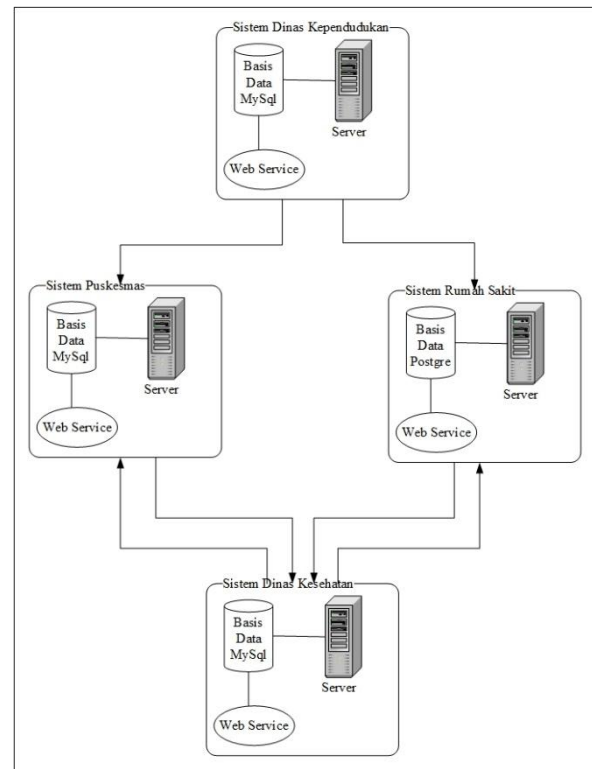
- c. Integrasi dan sinkronisasi antara sistem Rumah sakit dengan sistem Dinas Kesehatan dalam proses monitoring data laporan penyakit dan laporan penderita penyakit menggunakan *service* menggunakan web service SOAP dengan komunikasi data XML. Berikut adalah web service sistem Rumah sakit untuk memerikan data laporan penyakit dan laporan penderita penyakit.

1. getLaporanW1RS
2. getLaporanW2RS
3. getLaporanLB1RS
4. LaporanPenderitaPenyakitW1
5. LaporanPenderitaPenyakitW2
6. LaporanPenderitaPenyakitLB1

- d. Integrasi data ICD penyakit antara sistem Puskesmas dan sistem Rumah Sakit dengan sistem Dinas Kesehatan dibuat *service* untuk membagi data ICD penyakit ke sistem lain menggunakan *web service* SOAP dengan komunikasi data menggunakan XML. Berikut adalah web service yang diberikan oleh sistem Dinas Kesehatan.

1. getDataICD

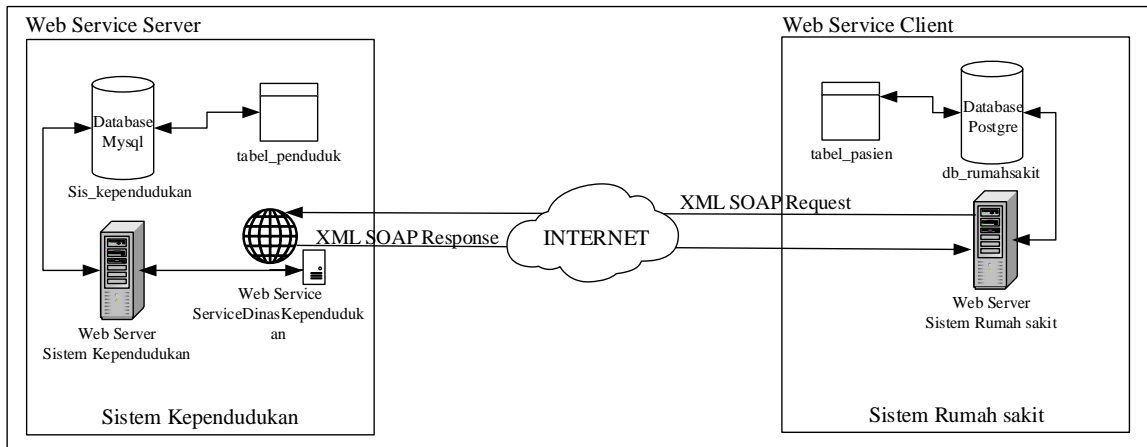
3.4 Rancangan Integrasi



Gambar 2: Desain Interaksi Web Service antar Sistem

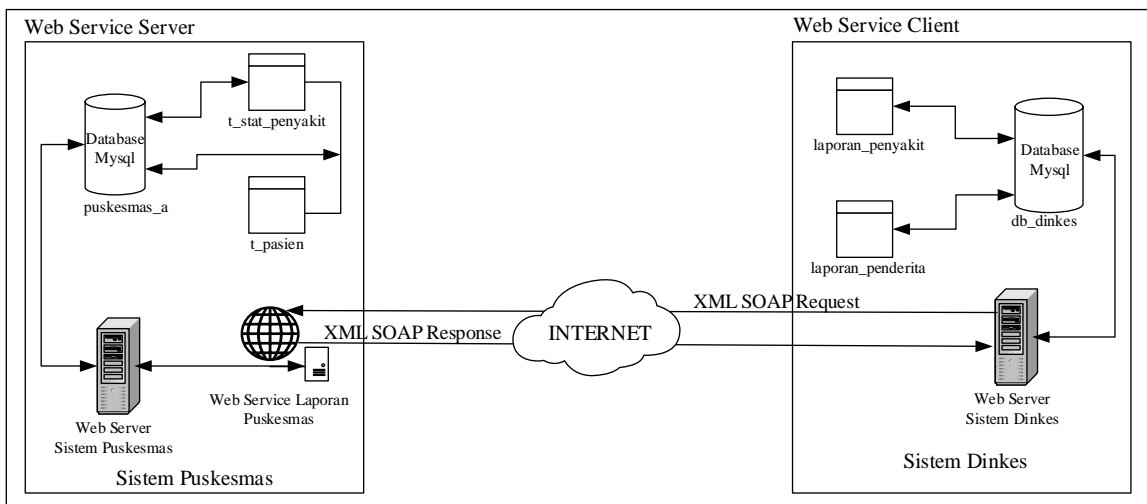
Penjelasan dari desain interaksi *web service* antar sistem yaitu sebagai berikut:

- a. Sistem Informasi Dinas Kependudukan



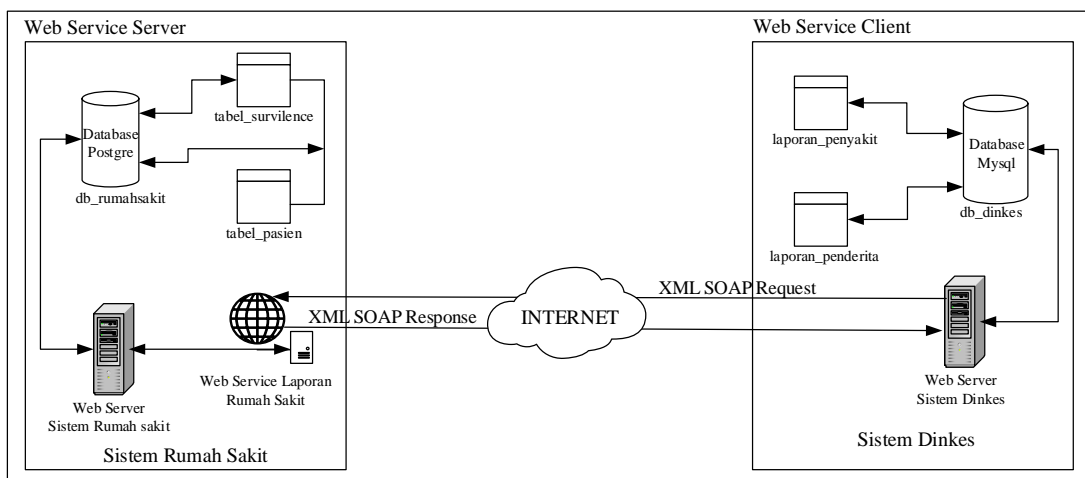
Gambar 3: Integrasi antara Sistem Kependudukan dengan Sistem Puskesmas

b. Sistem Informasi Puskesmas



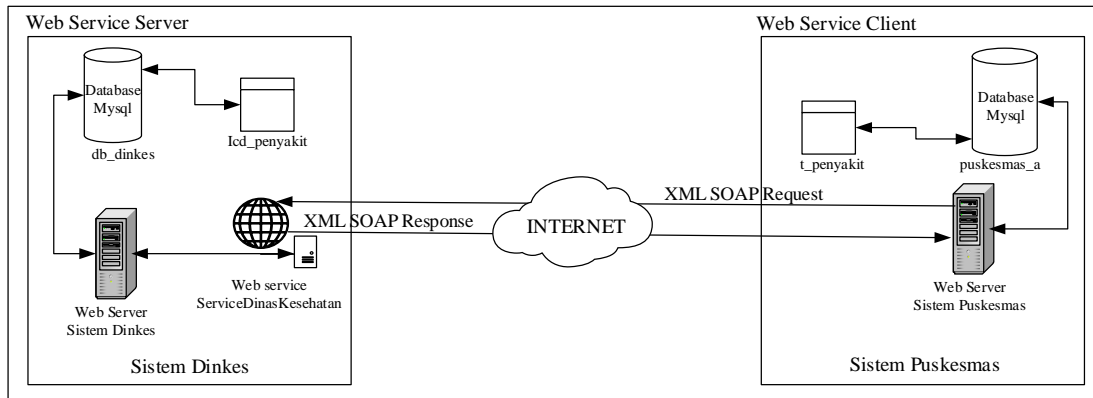
Gambar 4: Arsitektur integrasi Sistem Puskesmas dengan Sistem Dinkes

c. Sistem Informasi Rumah Sakit

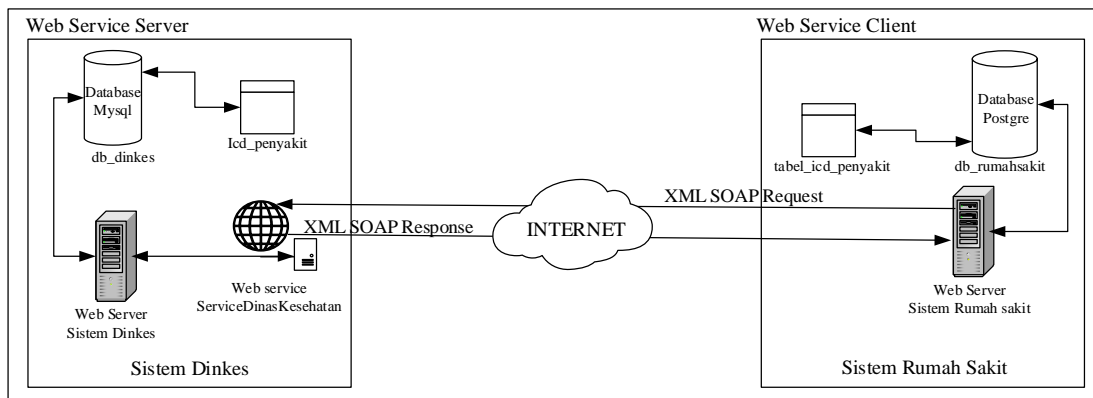


Gambar 5: Arsitektur integrasi sistem Rumah sakit dengan sistem Dinkes

d. Sistem Informasi Dinas Kesehatan



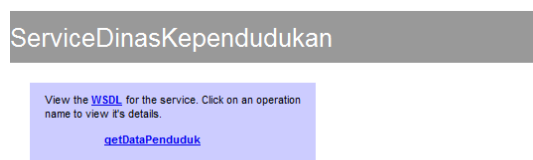
Gambar 6: Arsitektur Integrasi Sistem Dinas Kesehatan dengan Sistem Puskesmas



Gambar 7: Arsitektur Integrasi Sistem Dinas Kesehatan dengan Sistem Rumah sakit

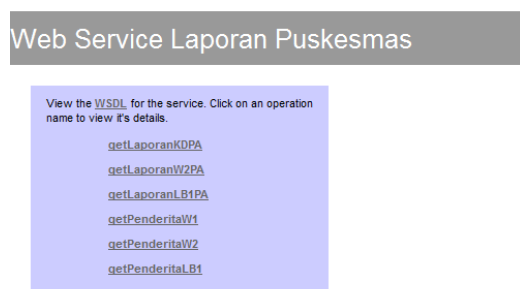
3.5 Implementasi

a. Sistem Informasi Dinas Kependudukan



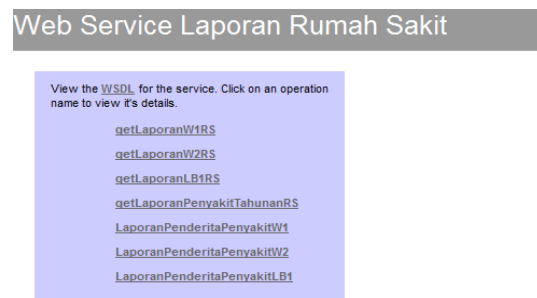
Gambar 8: Web service sistem Dinas Kependudukan

b. Sistem Informasi Puskesmas



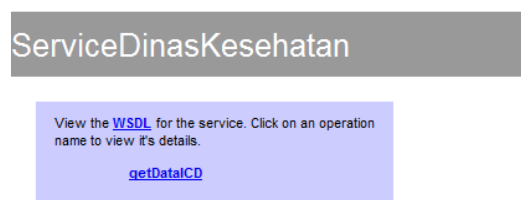
Gambar 9: Web service Sistem Puskesmas

c. Sistem Informasi Rumah Sakit



Gambar 10: Web service Sistem Rumah Sakit

d. Sistem Informasi Dinas Kesehatan



Gambar 11: Web service Sistem Dinkes

3.6 Pengujian

Pengujian akan dilakukan pada sistem integrasi laporan dan *web service* antar sistem yang berinteraksi, dari pengujian ini akan terlihat apakah data dari beberapa sistem yang terintegrasi memiliki data yang sama dan transfer data *web service* XML SOAP bisa berjalan dengan baik dengan teknik Black Box dan White Box.

a. Pengujian Integrasi Data

1

```
$tanggal=$tahun."-".$bulan."-".$tgl;
$url=$_SERVER['DOCUMENT_ROOT']
."/ta/dinkes/application/libraries/nusoap/lib/nusoap.php";
require_once("$url");
$client=new nusoap_client
("http://localhost/ta/puskesmasA/ws/puskesmas_service.php?ws
```

2

```
if(is_array($result)){
```

3

```
$jmlappuskesmas=count($result);
```

4

```
else{
    $jmlappuskesmas=0;
```

5

```
} //end if
```

6

```
$kdinstansi="PDR01";
// cek jumlah laporan
yang sudah masuk ke dbcenter
$kon=new PDO
("mysql:host=localhost;dbname=
db_dinkes",'root','');
$pr=$kon->prepare("SELECT
COUNT(*) AS jmlpenyakit FROM
`laporan_penyakit`WHERE
`tanggal_laporan`=:tanggal
AND
kode_instansi=:kodeinstansi");
$pr-
>bindParam(":tanggal",$tanggal);
$pr-
```

7

```
$jmllapdb=0;
}
```

8

```
else{
    $jmllapdb=$jmlr;
```

9

```
} //end if
```

10

```
if($jmlappuskesmas==0 &&
```

11

```
$hasil='nodata';
}
```

12

```
elseif ($jmlappuskesmas>0 &&
$jmllapdb==0) {
```

13

```
$hasil='nosinkron';
}
```

14


```
elseif ($jmlappuskesmas>0 &&
$jmllapdb>0 &&
$jmlappuskesmas==$jmllapdb) {
```

15

```
foreach ($result as $value) {
```

16

```
$pre=$kon->prepare("SELECT
jumlah_penyakit
FROM
laporan_penyakit
WHERE
kode_icd=:kdicd
AND
tanggal_laporan=:tgl
AND
kode_instansi=:kdinstansi");
$pre-
>bindParam(":kdicd",$value['ko
depenyakit']);
$pre-
>bindParam(":tgl",$tanggal);
$pre-
>bindParam(":kdinstansi",$kdin
stansi);
$pre->execute();
$res=$pre->fetchAll();
foreach ($res as $nil) {
$jmlpenyakitdb=$nil['jumlah_pe
nyakit'];
}
```

17

```
if($value['jmlpenyakit']!=$jml
penyakitdb){
```

18

```
$hasil="nosinkron2";
break;
}
```

19

```
else{
    $hasil='sinkron';
}
```

20

```
} //end foreach
```

21

```
elseif ($jmlappuskesmas>0 &&
$jmllapdb>0 &&
$jmlappuskesmas!=$jmllapdb) {
```

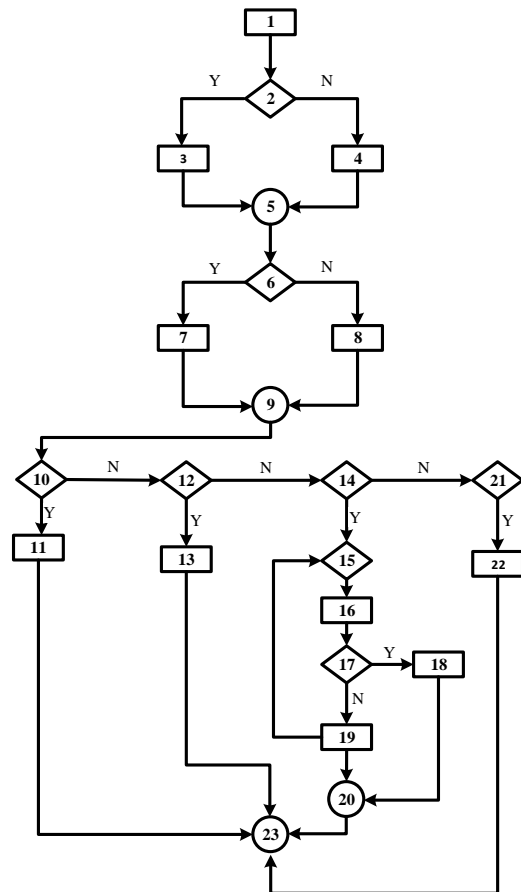
22

```
$hasil="nosinkron2";
}
```

23

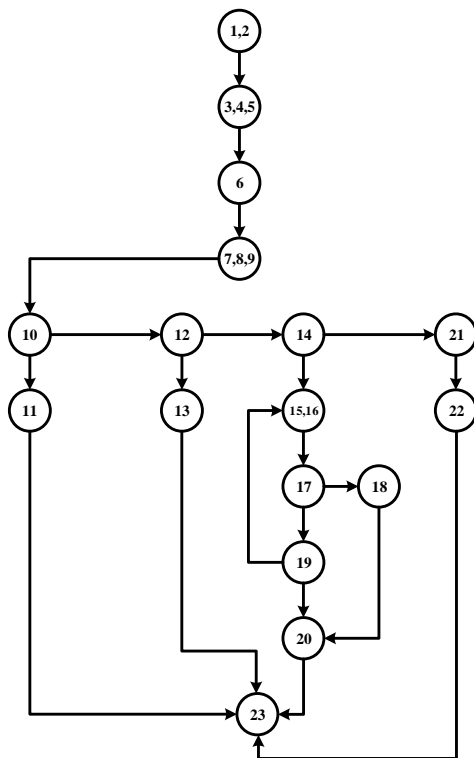
```
return $hasil;
```

b. Flow Chart Proses Sinkronasi



Gambar 12 Flow Chart Proses Sinkronasi

d. *Flow Graph* Proses Sinkronisasi



Gambar 13 *Flow Graph* Proses Sinkronisasi

Perhitungan *cyclomatic complexity* dari *flow graph* di atas didasarkan pada rumus:

$$V(G) = E - N + 2$$

Dari gambar sebelumnya diketahui:

Jumlah edge 'panah' = 21

Jumlah node 'titik' = 17

Sehingga didapatkan:

$$V(G) = 21 - 17 + 2$$

$$V(G) = 6$$

Dari persoalan di atas independent path yang diperoleh adalah sebagai berikut:

- 1,2 - 3,4,5 - 6 - 7,8,9 - 10 - 11 - 23
- 1,2 - 3,4,5 - 6 - 7,8,9 - 10 - 12 - 13 - 23
- 1,2 - 3,4,5 - 6 - 7,8,9 - 10 - 12 - 14 - 15,16 - 17 - 18 - 20 - 23
- 1,2 - 3,4,5 - 6 - 7,8,9 - 10 - 12 - 14 - 15,16 - 17 - 19 - 20 - 23
- 1,2 - 3,4,5 - 6 - 7,8,9 - 10 - 12 - 14 - 15,16 - 17 - 19 - 15,16 - 17 - 18 - 20 - 23
- 1,2 - 3,4,5 - 6 - 7,8,9 - 10 - 12 - 13 - 14 - 21 - 22 - 23

4. Hasil Penelitian

1. Integrasi Sistem Kependudukan dengan Sistem Rumah sakit

Hasil dari integrasi dari kedua sistem heterogen adalah data penduduk dapat digunakan oleh sistem Rumah sakit menggunakan *web service* SOAP dari sistem Kependudukan ke Sistem Rumah sakit dan hasilnya adalah data berhasil terintegrasi.

2. Integrasi Sistem Kependudukan dengan Sistem Puskesmas

Hasil dari integrasi dari kedua sistem heterogen adalah data penduduk dapat digunakan oleh sistem Puskesmas menggunakan *web service* SOAP dari sistem Kependudukan ke Sistem Puskesmas dan hasilnya adalah data berhasil terintegrasi

3. Integrasi Sistem Rumah sakit dengan Sistem Dinas Kesehatan

Hasil dari integrasi dari kedua sistem yang heterogen antara sistem Rumah sakit dengan sistem Dinkes dengan menggunakan *web service* SOAP untuk membagi data laporan dan data penderita penyakit berhasil terintegrasi. Serta data ICD penyakit yang ada di Dinkes berhasil terintegrasi dengan sistem Rumah sakit.

4. Integrasi Sistem Puskesmas dengan Sistem Dinas Kesehatan

Hasil dari integrasi dari kedua sistem yang heterogen antara sistem Puskesmas dengan sistem Dinkes dengan menggunakan *web service* SOAP untuk membagi data laporan pencatatan penyakit serta data penderita penyakit berhasil terintegrasi. Dan data ICD penyakit yang ada di Dinkes berhasil terintegrasi dengan sistem Puskesmas.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berikut adalah kesimpulan dari penelitian ini adalah sebagai berikut:

- Dengan memanfaatkan metode *web service* SOAP, maka integrasi data diantara sistem yang heterogen tidak perlu mengubah sistem yang sudah ada meskipun diantara sistem memiliki

konflik pada penggunaan DBMS yang berbeda, penamaan tabel yang berbeda dan struktur atribut tabel yang berbeda.

- b. Penggunaan *web service* sebagai solusi antar sistem yang heterogen lebih mudah karena tidak perlu mengubah sistem yang sudah ada meskipun ada konflik heterogenitas.

5.2 Saran

Untuk meningkatkan kinerja serta menyempurnakan sistem yang dibuat, penulis memberikan saran sebagai berikut:

- a. Untuk pengembangan perlu dibuat *service by demand* yaitu penggunaan *service* ketika diperlukan saja .
- b. Untuk integrasi dan sinkronisasi dilakukan secara *asynchronous* dengan program yang berjalan pada *background process* pada sistem operasi.
- c. Untuk perlu diadakan uji standar keamanan *web service* SOAP untuk pengamanan dokumen XML hasil komunikasi antara *server* dan *client*.

Referensi

[1] S. Potss and M. Kopack, Sams Teach Yourself Web Services in 24 Hours, Indiana: Sams Publishing, 2003.

[2] R. Saputra and A. Ashari, "Integrasi Laporan Demam Berdarah Dengue (DBD) Menggunakan Teknologi Web Service," Jurnal Masyarakat Informatika, vol. 2, no. 3, pp. 15-26, 2012.

[3] H. Deviana, "Penerapan XML Web service Pada Sistem Distribusi Barang," Jurnal Generic, vol. 6, no. 2, pp. 61-70, 2011.

[4] P. Swithinbank and dkk, Connecting Enterprise Applications to WebSphere Enterprise Service Bus, USA: IBM Corporation, 2007.

[5] P. L. Puustjärvi Juha, "Application Integration and Semantic Integration in Electronic Prescription Systems," IJCSI International Journal of Computer Science Issues, vol. 7, no. 2, pp. 1-8, 2010.

[6] S. S. Justin and S. T., "Trends and Issues in Integrating Enterprises and other Associated Systems using Web Services,"

Associated Systems using Web Services, vol. 1, no. 12, pp. 17-20, 2010.

[7] M. Kamal, "Exploring Knowledge Management Integration through EAI in Local Government Domain," in European, Mediterranean & Middle Eastern Conference on Information Systems 2010, Abu Dhabi, 2010.

[8] D. Risimic, "AN INTEGRATION STRATEGY FOR LARGE ENTERPRISES," Yugoslav Journal of Operations Research, vol. 17, no. 2, pp. 209-222, 2007.

[9] K. Nadezhda and dkk, "Integration Principles Of Russian And Japanese Databases On Inorganic Materials," International Journal "Information Technologies and Knowledge", vol. 2, pp. 366-372, 2008.

[10] Bunawan, "Pengantar Pengolahan Data Terdistribusi," Gunadarma, Jakarta, 2005.

[11] H. R. Sheikh, "Comparing CORBA and Web-Services in view of a Service Oriented Architecture," International Journal of Computer Applications, vol. 39, no. 6, pp. 47-55, 2012.

[12] H. R. Sheikh, "Comparing CORBA and Web-Services in view of a Service Oriented Architecture," International Journal of Computer Applications, vol. 39, no. 6, pp. 47-55, 2012.

[13] R. Richards, APress Pro PHP XML and Web Services, New York: Appres, 2006.

[14] R. Daniel, "Pengenalan Konsep XML Web Services," [Online]. Available: <http://ikc.dinus.ac.id/populer/roy/roy-webservices.pdf>. [Accessed 10 Agustus 2014].

[15] M. Junaedi, "IlmuKomputer," Agustus 2006. [Online]. Available: <http://www.ilmukomputer.org/wp-content/uploads/2006/08/junaedi-xml.zip>. [Accessed 8 10 2014].

[16] K. Wagh and D. R. Thool, "A Comparative Study of SOAP Vs REST Web Services Provisioning Techniques for Mobile Host," Journal of Information Engineering and Applications, pp. 12-17, 2012.