DESIGN RESTFUL WEB SERVICE OF NATIONAL POPULATION DATABASE FOR SUPPORTING E-HEALTH INTEROPERABILITY SERVICE

By Deris Stiawan

WORD COUNT 5576 TIME SUBMITTED 12-MAR-2020 10:52AM
PAPER ID 56387083

Journal of Theoretical and Applied Information Technology
15th August 2018. Vol.96. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

DESIGN RESTFUL WEB SERVICE OF NATIONAL POPULATION DATABASE FOR SUPPORTING E-HEALTH INTEROPERABILITY SERVICE

[1]M. MIFTAKUL AMIN, [2]ADI SUTRISMAN, [3]DERIS STIAWAN, [4]ERMATITA, [5]ANDINO MASELENO

[1, 2] Department of Computer Engineering, Politeknik Negeri Sriwijaya, Palembang, Indonesia

[3,4] Department of Computer Science, Universitas Sriwijaya, Palembang, Indonesia

[5] Department of Information System, STMIK Pringsewu, Lampung, Indonesia

E-mail: [1] mmiftakulamin@gmail.com; miftakul_a@polsri.ac.id, [2] adistra74@gmail.com, [3]deris@unsri.ac.id, [4] ermatitaz@yahoo.com, [5] andimaseleno@gmail.com

ABSTRACT

The Government of the Republic of Indonesia nowdays has implemented a centralized national population database with Electronic Identity Card (e-ID) as the sole reference to the legality and validity of data used for various public services. In line with the population database, the government has also run a National Health Insurance Program which also needs e-ID card data to obtain health services. When patients visit health service providers such as hospitals, health centers, medical centers, and health clinics, personal data is requested for registration by showing ID cards, so the presence of a national demographic database is not yet optimal. The presence of RESTful Web Service technology enables communication between different platform applications, so the demographic database can be accessed using a variety of different client applications, operating systems, programming languages, and databases. This study aims to design the national demographic database using RESTful Web Services, which can be implemented to support e-health services. E-ID data can be used by the health provider for patient registration and patient referral. The demographic database can be used as a reference for valid demographic data. The test that has been done shows that the performance of the system design developed, reliable to be implemented with the accuracy of the acquisition of data records of valid residence in accordance with the design of data structures.

Keywords: *Population Database, E-Id, Interoperability, Restful Web Services, E-Health*

1. INTRODUCTION

The emergence of a variety of population problems has encouraged Indonesian government to apply the electronic population system, as a national population database system solution for every citizen. This population database will be a single reference, so it is designed centrally to avoid duplication of population data. To overcome the problem of population data duplication, the Indonesian government applies e-ID card to replace conventional ID card [1]. Each agency, whether government or private, can use a population database as a reference to the legality and validity of data of every Indonesian citizen.

The Government of Indonesia through the Ministry of Home Affairs is implementing e-ID card program based on Population Identity Number (PIN) as a form of implementation of Law no. 23 of 2006 on Population Administration [2]. In line with other Indonesian government programs the National Health Insurance Program organized by the Social Security Administering Agency has been officially applied to all Indonesian citizens since January 2014 [3]. This encourages the need for integration of population data in the National Health Insurance Program, as outlined in the Presidential Instruction of the Republic of Indonesia no. 8 year 2017 on optimizing the implementation of the National Health Insurance Program. Furthermore, the Minister of Home Affairs affirmed that the Inpres must be implemented by all Regional Heads.

Various government services will substitute e-ID cards as a single identity (with certain access permits) for every Indonesian citizen [4]. Services such as passport making, tax payments, drivers licensing, health services,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

insurance, land certificate issuance, and various other services using e-ID card [5].

The health sector is one of Indonesia's 10 national priorities as the foundation for all projects and programs run by the Indonesian government in 2018. The need for e-ID card data is a key reference in e-health services for health care providers such as hospitals, community health center, clinics, health clinics, and social security administering agency is a new challenge in the development of health information systems.

In Indonesia, the Ministry of Health has formulated the National Health System in 2004, which has two main elements of health efforts covering, community health and individual health efforts [6]. In the implementation of community health efforts can be realized by building at least one community health center in each sub-district, and supported by agencies in districts/cities and provinces. While the individual health efforts is realized through the practice of doctors, midwives, polyclinics, clinics, maternity homes, specialist clinics, and hospitals, both government and private. The network of the service system requires an interconnected and related health information system (e-health).

E-ID data has not been fully utilized in e-health services, since communication has not been done at the application level between health care providers and national demographic databases. In addition, each of the health service providers have used information systems with heterogeneous platforms viewed from operating systems, programming languages, and databases used. This research is one of the efforts to provide a software development architecture capable of supporting the interoperability of e-health services in providing e-ID based on demographic database.

The purpose of this research are:

- Identify RESTful web service as integration model to integrate national population database and e-health application services.
- Validate the integration model through implementation testing using software testing tools and client application testing.

The significances of the research are:

- The findings of this research can be used to prove that technology of RESTful web service can be implemented in a distributed system environment to realize interoperability between applications.
- It can be used as an effort to develop knowledge and technology especially in the field of service oriented architecture using RESTful web service.

This research is expected to contribute to the integration method technique using RESTful web service, how the integration process can be done, how the right data format can be exchanged, decomposition of business process in a service that can help understanding programmer in the stages of making program code.

2. LITERATURE REVIEW

Research with the topic of web service has been widely conducted and implemented in various fields. Research conducted by Adadi [7] has done the composition of web services in e-government applications, to bridge cross-platform applications between government agencies in Morocco. Web services are also implemented in finance to build billing and accounting systems [8]. This study is based on heterogeneous conditions and requirements related to the provision of different layouts, so that this web service technology can be used to automate. In the field of health, web services are used to perform electronic data exchange standards [9]. This research has applied web services technology to integrate Electronic Medical Records data in heterogeneous environment and create efficient information system.

In the field of education the implementation of web services is used in the new student enrollment system [10]. This research uses object oriented analysis and design model. Based on internal and external testing that has been done, found that the implementation of software using web services runs well.

Implementation of web services in the field of vacancies has also been implemented to inform relevant information for job seekers [11]. Using the method of combining semantic matching and simple additive weighting (SAW) algorithms, it shows that the web service can present the best relevant data and found based on the calculation of the similarity of functional parameters and service quality.

By using the SOAP protocol, Amran [4] has implemented web services for distributed electronic ID card systems. By using the Java programming language and Visual Basic. NET has developed a web service, so that electronic ID card data can be accessed for various purposes.

Hassan [12] in his research builds a framework of individual level technology adoption

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

known as I-P (Individual Privacy) that integrates Smart National Identity Card (SNIC) with health information systems. This study combines methods of Unified Theory of Acceptance and Use Technology (UTAUT) and the Privacy Calculus Model (PCM). This research can be used by governments to plan development, such as building e-government, decision-making processes and public policy, and other government projects.

The research will be to build a proposed distributed web services based system using REST protocol to provide population database and can be used by various client application platforms, to support e-health service.

The motivation of this research is that, currently no data exchange mechanism between national population database and health information systems as e-health services. This research is important to be done in optimizing e-health services, to integrate data between different organizations required communication bridge by using web service technology. With this technology, it allows to provide a separate set of Application Programming Interfaces (APIs) with applications that will access the API. RESTful web service is selected, based on its advantages. After a set of APIs is completed, the test stage can be done with a module driven testing approach to ensure that the API works properly. This testing stage will use the help of an application that can provide performance information for each API that has been developed.

3. THEORY BACKGROUND

3.1 Service Oriented Architecture (SOA)

Service Oriented Architecture (SOA) technology refers to the design of distributed system. It is an architecture methodology and design that uses a variety of services that are implemented on multiple platforms and various programming languages [13]. SOA provides a set of interfaces to enable clients to consume the services provided [14]. SOA involves 3 main parts, namely service providers, service consumers and service brokers who run 3 main functions, namely publish, find, and bind [15]. SOA is basically a set of services [16]. Each service communicate with each other as illustrated in figure 1. The service is a well-defined implementation of business functions, and the service can then be consumed by clients in various applications or business processes.

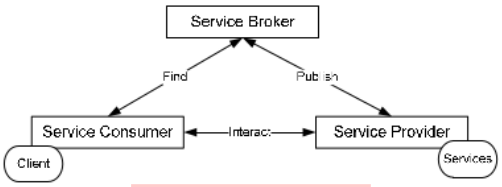

Figure 1: SOA Architecture

3.2 Interoperability

Interoperability within the IEEE Standard Computer Dictionary is defined as the ability of two or more systems to exchange data or information and mutual use of data or information exchanged [17]. In the early development of distributed applications to achieve interoperabilty available a variety of technologies such as RMI, CORBA, and DCOM as communication between client and server [18]. The limitation of such technology is on the dependency aspect (platform dependency) used. As RMI can only run on Java platform, while DCOM uses Windows platform. Another limitation is the aspect of compatibility and security when implementing it. Next comes web service technology as an independent platform, allowing applications to communicate in heterogeneous environments.

3.3 Role of Web Services in Distributed Computing

As one of the distributed system technology, web services can be accessed using the HTTP protocol over the Internet network, so that web services can be built and accessed using a variety of programming languages [19]. In distributed systems there are several layers, namely presentation layer, business layer, and data access layer as shown in Figure 2. The client side is a presentation layer that can be enterprise application, desktop application, mobile application, web application, or lightweight application type such as web browser. Next through the SOAP protocol and REST client invoke to request a message to the web service in which there is a business layer.

Figure 2: Web Service Usage

Web service is located on remote servers accessed through the Internet, and is responsible for providing response messages to clients that request them. The data exchange format can be performed using several types of data formats such as XML,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

HTML, RSS, Atom, CSV, JSON and other formats [20]. XML and JSON are data formats that are widely used in web services development. If the information presented in the form of a web site can be read directly by humans, then the information presented in web services, will be accessed by the software, so it is not directly consumable by humans [21].

3.4 Restful Web Services

REST (REpresentational State Transfer) is a standard web-based communication architecture that is often applied in the development of web service. Web Service is one technology that can be used to perform data integration between information systems [22]. Generally the use of HTTP (Hypertext Transfer Protocol) as a protocol for data communication. REST was first introduced by Roy Fielding in 2000. In REST architecture, REST server provides resources (resources / data) and REST clients access and display the resource for future use. Each resource is identified by URIs (Universal Resource Identifiers) or global ID [23]. The resource is represented in the form of text format, JSON or XML. In general the format used is JSON and XML.

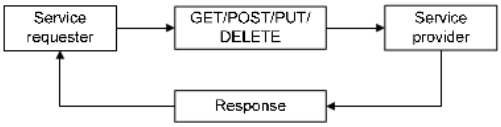

Figure 3: RESTful Web Service Architecture

Figure 3 provides an overview of RESTful Web Service architecture as a realization of the SOA concept. This architecture uses mechanisms such as how web applications work, by using HTTP methods, stateless, exposing directory structures using URLs, and transferring data in XML and JSON formats. Some of the HTTP methods used in RESTful Web Service are:

- GET, used to obtain data from a resource.
- POST, used to add new resources.
- PUT, used to modify a resource.
- DELETE, used to remove a resource.

3.5 SOAP and REST protocols

There are two main protocols in the development of web services, namely SOAP and REST protocol [24, 25, 26, 27]. Table 1 shows the comparison between SOAP and REST protocols.

Table 1: Difference Between SOAP and REST

SOAP	REST
Designed to be used in a distributed environment	Designed for use on point-to-point communications.
Known as traditional technology.	Considered as the new technology successor of SOAP.
Communication between client and server is tighly coupled	Communication between the client and server is loosely coupled which means the design is made freely and independent, so it is easily changed and more flexible.
Changes on the server side will greatly affect the changes that exist on the client side	Not too many changes to be made on the client side, if there is a change on the server side
The number of data transferred from server to client is quite large	The number of data transferred from server to client is light enough
SOAP always returns data in XML.	The data returned is quite flexible.
Response time is quite high.	Response time is quite low.

4 PROPOSED APPROACH

4.1 Software System Model

The architecture of the RESTful Web Service software to provide the national population database system that will be designed can be shown in Figure 4. To provide service for client applications, a system designed to implement a web application interface as an interaction layer between client and server. Using HTTP Methods such as GET, POST, PUT, and DELETE web services can serve requests from the client side and return the response in JSON format. The software used to build RESTful Web Service in this research is the PHP programming language, whereas to accommodate the database persistently using the MySQL DBMS. For a RESTful Web Service trial, use Apache as a Web Server.

Journal of Theoretical and Applied Information Technology
15th August 2018. Vol.96. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

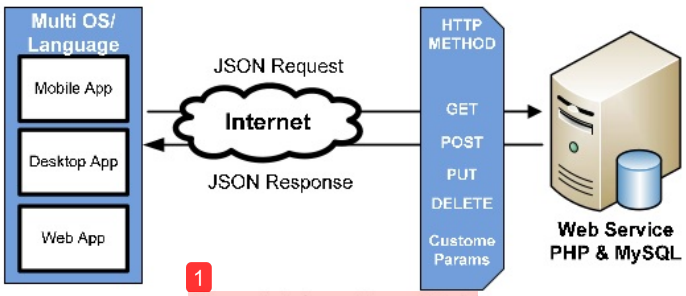

Figure 4: Software System Model

On the client side can use a variety of application platforms written using different programming languages, and different operating systems as well. On the client side can use mobile applications, desktop applications, and web applications to access the web services database demographics. Data formats are exchanged between client and server using JSON data format.

4.2 Physical Database Diagram

Database developed using physical database model diagram as can be seen in Figure 5. Table Electronic Identity Card (EID) using Population Identity Number (PIN) as the sole identity of citizenship. For domicile information, then provided table country, province, regency/district, subdistrict, and village.

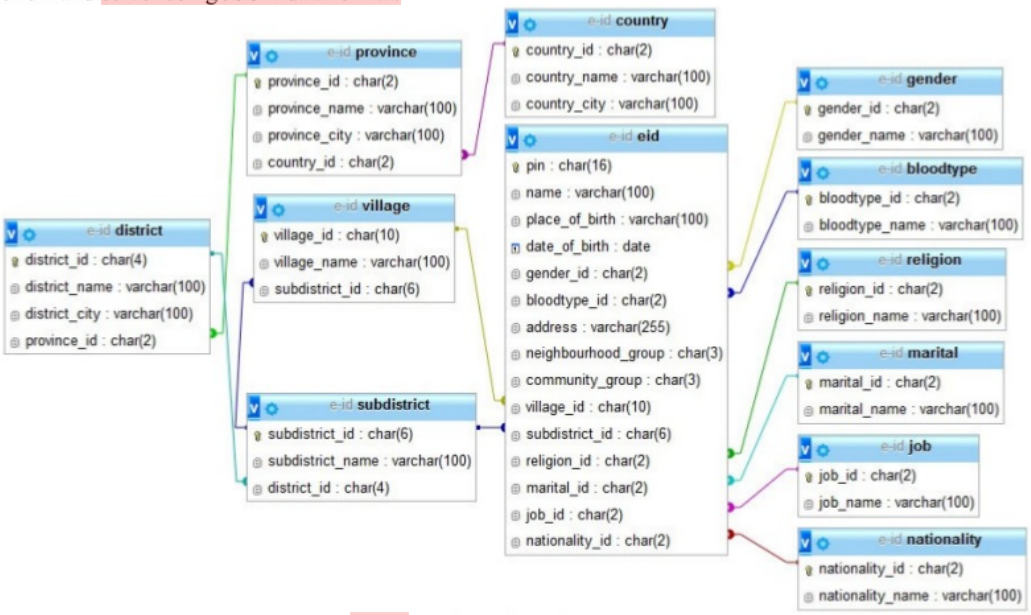

Figure 5: Physical Database Diagram

To accommodate individual profile information of each citizen as master data provided table gender, bloodtype, religion, marital, job, and nationality. So that detailed information of every citizen can later be presented completely in the developed system.

5. RESULTS AND DISCUSSION

In its implementation, Restful web service is placed on a web server, and uses an Internet connection to access the service. Through the HTTP method GET, POST, PUT, and DELETE. The Services of Web service can be consumed with various types of applications built using a variety of programming languages. To use the service, URI addresses are used as routing.

5.1 Routing Restful Web Services

Design of the restful web service routing can be seen in Table 2. There are 60 URL addresses provided by the system. Each URL address is a representation of the method that serves as services.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: List of Web Service's Method Routing

No.	RESTful URL	HTTP Method	Description
1.	/listCountry	GET	Service for dispalying country list
2.	/findCountryById/{country_id}	GET	Service for searching country data based on country_id field.
3.	/insertCountry	POST	Service for adding new data into country table.
4.	/updateCountry	PUT	Service for updating country data
5.	/deleteCountry	DELETE	Service for deleting country data
6.	/listProvince	GET	Service for displaying province list.
7.	/findProvinceById/{province_id}	GET	Service for searching data province based on province_id field.
8.	/insertProvince	POST	Service for adding new data into province table.
9.	/updateProvince	PUT	Service for updating province data
10.	/deleteProvince	DELETE	Service for removing province data
11.	/listDistrict	GET	Service for displaying district list
12.	/findDistrictById/{district_id}	GET	Service for searching district_id field.
13.	/insertDistrict	POST	Service for adding new data into table.
14.	/updateDistrict	PUT	Service for updating district data
15.	/deleteDistrict	DELETE	Service for deleting district data.
16.	/listSubdistrict	GET	Service for displaying subdsitrcit list
17.	/findSubdistrictById/{subdistrict_id}	GET	Service for looking for subdistrict data based on subdistrict_id field.
18.	/insertSubdistrict	POST	Service for adding new data into subdistrict table.
19.	/updateSubdistrict	PUT	Service for updating subdistrict data.
20.	/deleteSubdistrict	DELETE	Service for removing subdistrict data.
21.	/listVillage	GET	Service for displaying village list
22.	/findVillageById/{village_id}	GET	Service for using village data based on village_id field.
23.	/insertVillage	POST	Service for adding new data into village table.
24.	/updateVillage	PUT	Service for updating village data.
25.	/deleteVillage	DELETE	Service for removing village data.
26.	/listGender	GET	Service for displaying gender list.
27.	/findGenderById/{gender_id}	GET	Service for searching gender data based on gender_id field.
28.	/insertGender	POST	Service for adding new data into gender table.
29.	/updateGender	PUT	Service for updating gender data.
30.	/deleteGender	DELETE	Service for deleting gender data.
31.	/listBloodtype	GET	Service for displaying bloodtype list.
32.	/findBloodtypeById/{bloodtype_id}	GET	Service for searching bloodtype data based on bloodtype_id field.
33.	/insertBloodtype	POST	Service for adding new data into table.
34.	/updateBloodtype	PUT	Service for updating bloodtype data.
35.	/deleteBloodtype	DELETE	Service for removing bloodtype data.
36.	/listReligion	GET	Service for displaying religion list.
37.	/findReligionById/{religion_id}	GET	Service for searching religion data based on religion_id field.
38.	/insertReligion	POST	Service for adding new data into religion table.
39.	/updateReligion	PUT	Service for updating religion data.
40.	/deleteReligion	DELETE	Service for deleting religion data.
41.	/listMarital	GET	Service for displaying marital list.
42.	/findMaritalById/{marital_id}	GET	Service for searching marital data based

			on marital_id field.
43.	/insertMarital	POST	Service for adding new data into marital table.
44.	/updateMarital	PUT	Service for updating marital data.
45.	/deleteMarital	DELETE	Service for deleting marital data.
46.	/listJob	GET	Service for displaying job list.
47.	/findJobById/{job_id}	GET	Service for searching job data using job_id field.
48.	/insertJob	POST	Service for adding new data into job table.
49.	/updateJob	PUT	Service for updating job data.
50.	/deleteJob	DELETE	Service for deleting job data.
51.	/listNationality	GET	Service for displaying nationality list.
52.	/findNationalityById/{nationality_id}	GET	Service for searching nationality data based on nationality_id field.
53.	/insertNationality	POST	Service for adding new data into nationality table.
54.	/updateNationality	PUT	Service for updating nationality data.
55.	/deleteNationality	DELETE	Service for deleting nationality data.
56.	/listEid	GET	Service for displaying the electronic identitiy card (EID).
57.	/findEidById/{pin}	GET	Service for searching data based on population identification number (PIN) field.
58.	/insertEid	POST	Service for adding new data into EID table.
59.	/updateEid	PUT	Service for updating EID data
60.	/deleteEid	DELETE	Service for deleting EID data.

5.2 Testing using Postman Software

Postman become one of the important tools to do web services testing. There are some other popular software used, namely SOAP UI and HTTP Master. In this research postman software is chosen because of ease of use and simple in its operation. Postman can be used by both developers and users to test the API that has been developed. This software supports web service testing, that using SOAP and REST protocol. RESTful web service testing is conducted to ensure the APIs that has been developed runs properly, before being integrated into the real world application.

Postman software is used to perform testing service that has been designed as shown in the routing in Table 2. By entering the URL address, and entering parameters on the body, it can be done the request and response process as shown in Figure 6. Data formats are exchanged on the request process/response using Javascript Object Notaion (JSON). Of the 60 services that have been designed, everything is going well and delivering performance as expected.

As shown in Figure 6, Postman supports HTTP Methods like GET, POST, PUT, DELETE and so on. It provides an interface for invoking RESTful web services and showing its results. It also displays the status codes to provide status information of the executed operation. For example the status code 200 means successful invocation of web service and results has been returned by web service in JSON format.

Journal of Theoretical and Applied Information Technology
15th August 2018. Vol.96. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

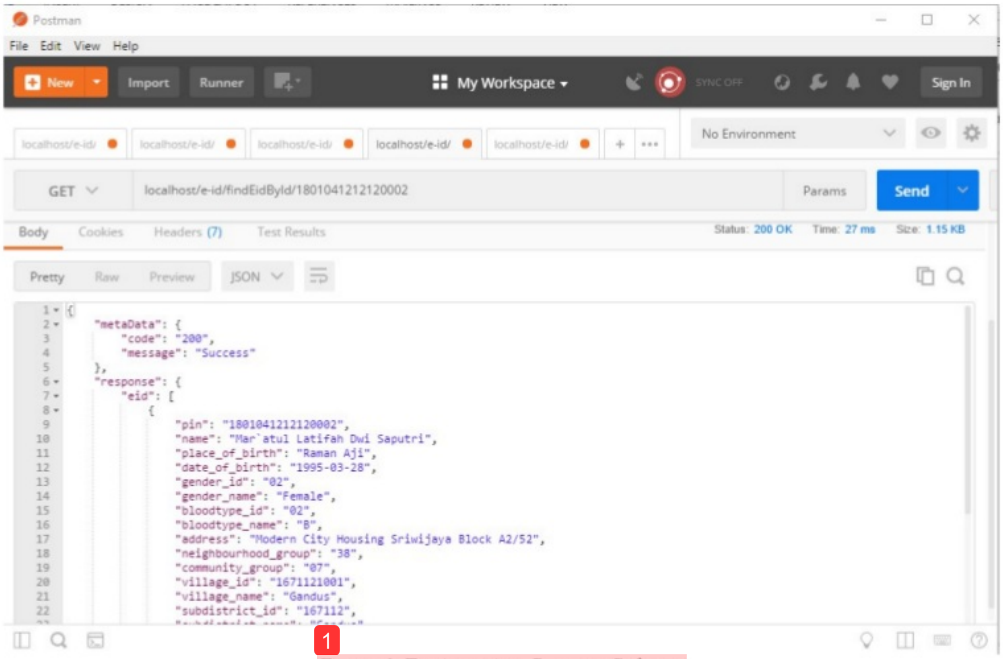

Figure 6: Testing using Postman Software

5.3 Testing using Desktop Application

RESTful web service URL can be used by hospitals, health centers, health clinics, and clinics to collect patient data based on Population Identification Number (PIN). Figure 7 shows the desktop application using the RESTful web service service, and displays the E-ID details data according to the information stored in the residence database.

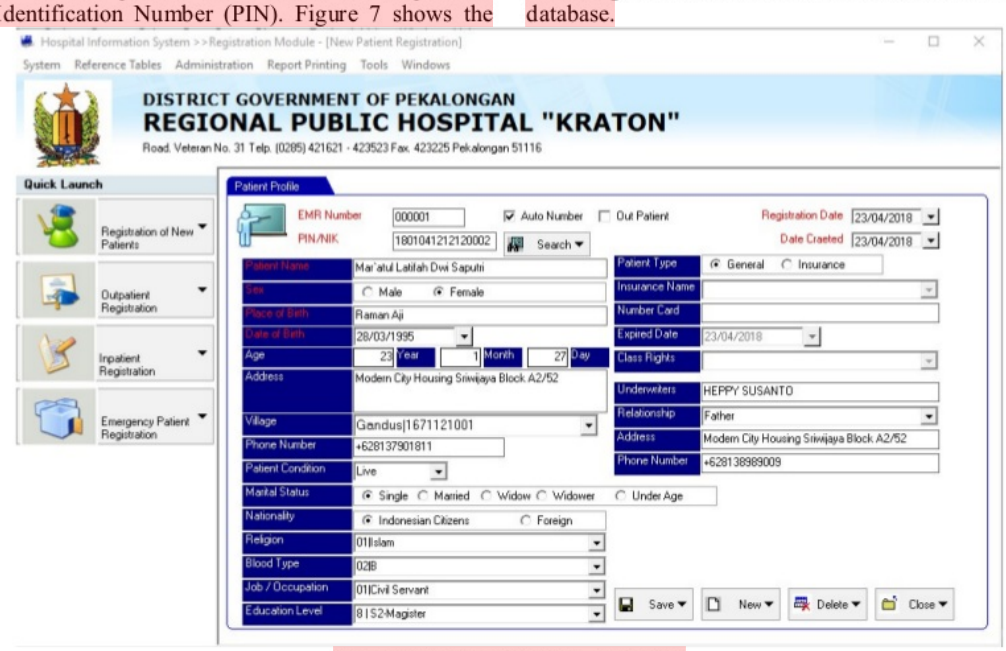

Figure 7: Testing of Desktop Application

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5.4 Performance RESTful Web Service

Table 3 shows the comparison of client access times to methods/services provided by RESTful web services, both local and online. From the performance test of RESTful web service, both from local and online are obtained different results in time. Some factors that affect the performance of RESTful web service among others :

1. **Web server processing speed**, it is possible that computing resources are used by other processes that occur within the computer server.

2. **Network bandwidth**, this is because the amount of bandwidth is different when executing a web service method. It is seen that, the time required to access the RESTful Web Service placed online is greater than the execution time of the local.

3. **Payload**, the amount of data or records that are transported between client/server. It is seen that, service involving POST, PUT, and DELETE process is bigger the time of execution because besides data sent to server, server also send data as response for client.

Table 3: List of Test Results RESTful Web Services (milli scond)

No.	Services	local	online
1.	/listCountry	19	237
2.	/findCountryById/ {country_id}	49	213
3.	/insertCountry	128	349
4.	/updateCountry	158	394
5.	/deleteCountry	137	203
6.	/listProvince	54	250
7.	/findProvinceById/ {province_id}	56	201
8.	/insertProvince	177	477
9.	/updateProvince	148	307
10.	/deleteProvince		
11.	/listDistrict	53	379
12.	/findDistrictById/ {district_id}	80	408
13.	/insertDistrict	133	758
14.	/updateDistrict	94	499
15.	/deleteDistrict	88	324
16.	/listSubdistrict	63	376
17.	/findSubdistrictById/ {subdistrict_id}	72	566
18.	/insertSubdistrict	130	331
19.	/updateSubdistrict	99	583
20.	/deleteSubdistrict	111	450
21.	/listVillage	52	368
22.	/findVillageById/ {village_id}	81	487
23.	/insertVillage	103	528
24.	/updateVillage	119	445
25.	/deleteVillage	92	362
26.	/listGender	61	428
27.	/findGenderById/ {gender_id}	71	334
28.	/insertGender	130	341
29.	/updateGender	135	493
30.	/deleteGender	117	450
31.	/listBloodtype	54	382
32.	/findBloodtypeById/ {bloodtype_id}	56	371
33.	/insertBloodtype	131	525
34.	/updateBloodtype	161	518
35.	/deleteBloodtype	141	480
36.	/listReligion	45	206
37.	/findReligionById/ {religion_id}	55	345
38.	/insertReligion	118	501
39.	/updateReligion	91	242
40.	/deleteReligion	162	185
41.	/listMarital	36	154
42.	/findMaritalById/ {marital_id}	60	225
43.	/insertMarital	139	300
44.	/updateMarital	102	264
45.	/deleteMarital	85	275
46.	/listJob	64	200
47.	/findJobById/ {job_id}	73	301
48.	/insertJob	147	441
49.	/updateJob	121	356
50.	/deleteJob	117	297
51.	/listNationality	17	124
52.	/findNationalityById/ {nationality_id}	26	130
53.	/insertNationality	128	326
54.	/updateNationality	115	480
55.	/deleteNationality	137	366
56.	/listEid	74	293
57.	/findEidById/{pin}	46	443
58.	/insertEid	111	764
59.	/updateEid	127	557
60.	/deleteEid	74	527

Figure 8 is a graphical display that describes the average execution process of each service provided. From the picture shows that the speed of web services processing online is more time-consuming than the local web service processing.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

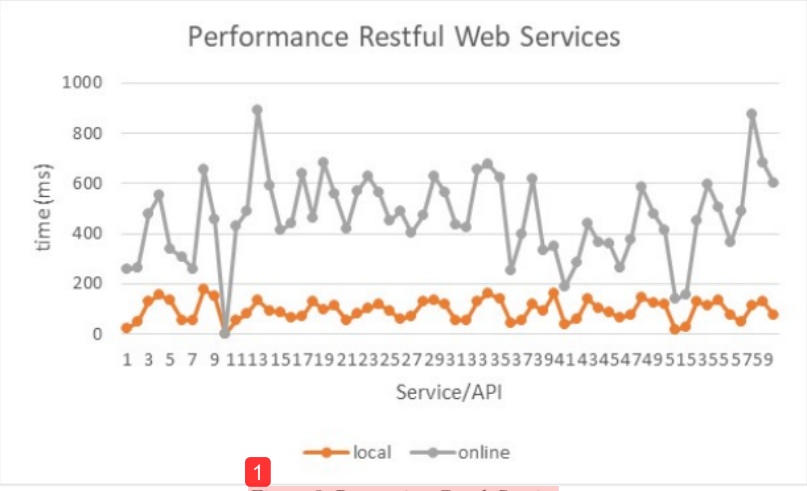

Figure 8. Processing Graph Service

Based on the results of research that has been done, obtained some information about the characteristics of RESTful web service development, including:

- We can develop a complex system by splitting it into several methods or procedures that serve as a service.
- The Application Programming Interface (API) that acts as a service can work like Create, Read, Update, and Delete (CRUD) operations using HTTP method, such as POST, GET, UPDATE, and DELETE. So it can perform data exchange operations, and manipulation of data directly related to the database management system (DBMS).
- RESTful web services can serve as back-end services, and serve a variety of client applications running on different operating systems, software, and programming languages.
- Web service development requires identification resources, so it needs good documentation and complete, so that the service provided can be accessed by client application. Its consequence is must to create a routing table containing the URL address information, input and output parameters, and the HTTP method used as presented in Table 2.

The performance test of the web service in this research, only in the phase of suitability of data structures generated by API with Postman software. With this software that acts as software testing tool can see the execution speed of the API that has been developed. It is possible to conduct research in an effort to improve the performance of RESTful web services, such as speed up the execution process, web service composition, and various other efforts to improve the performance of RESTful web service applications.

6. CONCLUSIONS

This research has implemented 60 methods as Application Programming Interface (API), which can be accessed through client application using HTTP Method GET, POST, PUT, and DELETE. Restful web services are easy to develop and distribute, accessible through a variety of terminals such as laptops, personal computers (PCs), and smartphones.

Web-based Distributed Services system using REST protocol is one of the reliable alternative technologies in building cross platform applications, and information system integration. The e-health services can be enhanced by the availability of e-ID data as the national population database referral data, and can be accessed directly through communication between software (applications) without manual processes involving humans. Based on the tests that have been done, the performance of RESTful web service is influenced by the speed of web server processing, network bandwidth, and payload.

This research has not discussed about security and privacy aspects in RESTful web service development. This research focuses on designing data structures, designing interfaces, data exchange mechanisms, and realizing application interoperability between national population databases and e-health services applications.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REFERENCES:

[1] Kurniati, P.S., "Policy Implementation of Electronic Identity Card (E-ID) in Cimahi City, West Java Province, Indonesia", *International Journal of Science and Research (IJSR)*, Vol. 4, No. 2, 2015, pp. 31-41

[2] Sutanta, E. and Ashari, A., "Pemanfaatan Database Kependudukan Terdistribusi Pada Ragam Aplikasi Sistem Informasi di Pemerintah Kabupaten/Kota", *Jurnal Ilmiah SISFOTEKNIKA*, Vol. 2, No. 1, 2012, pp. 11-20.

[3] Hariyanto, S., Nugroho, E., and Fauziati, S., "Model Integrasi Simpus Dalam Skema BPJS Kesehatan", *Proceedings CITEE 2017*, pp. 218-225.

[4] Amran, R.Y., "Interoperabilitas Sistem KTP Elektronik Terdistribusi Berbasis Simple Object Access Protocol (SOAP)", *Jurnal Inspiration*, Vol. 2, No. 1, 2012, pp. 67-84.

[5] Kurniati, P., S., Rusli, B., Muhtar, E.A., Sintaningrum, "Effectiveness of Evaluation of Electronic Identity Card (E-ID) Policy in Cimahi City, West Java Province, Indonesia", *Global Journal of Political Science and Administration*, Vol. 3, No. 5, 2015, pp. 21-31.

[6] Paturusi, S.D.E, "Ekstraksi Servis pada Sistem Informasi Kesehatan: Sebuah Pendekatan Ke Arah SOA (Service Oriented Architecture)", *Jurnal Tekno*, Vol. 8, No. 53, 2010, pp. 82-90

[7] Adadi, A., Berrada, M., Chenouni, D., Bounabat, B., "A Semantic Web Service Composition for E-Goverment Services", *Journal of Theoretical and Applied Information Technology*, Vol. 17, No. 3, 2015, pp. 460-467.

[8] Muthulakshmi, Anand, S., "Web Services Billing And Accounting", *Journal of Theoretical and Applied Information Technology*, Vol. 65, No. 1, 2014, pp. 121-129.

[9] Batra, U., Mukherjee, S., Nandal, A., Sachdeva, S., "Explorative Study of Healtcare Data Interchange Standards", *Journal of Theoretical and Applied Information Technology*, Vol. 63, No. 2, 2014, pp. 420-430.

[10] Utomo, W.H., Deviana, S., "XML Web Service Based Integration Modeling of New Student Admission in Junior High School", *Journal of Theoretical and Applied Information Technology*, Vol. 54, No. 3, 2013, pp. 535-540.

[11] Susyanto, T., Zuhrah, "Discovery and Integration of Job Market Service Using Semantic Web Service Approach", *Journal of Theoretical and Applied Information Technology*, Vol. 95, No. 2, 2017, pp. 412-421.

[12] Hassan, I.B., "Acceptance of Health Information Application in Smart National Identity Card (SNIC) Using a New I-P Framework", *Word Academy of Science, Engineering and Technology*, Vol. 8, No. 12, 2014, pp. 3782-3790.

[13] Alboaie, L., Buraga, S., "Service Oriented Architecture for Health System", 1st *National Symposium on e-Health and Bioengineering (EHB),* EHB 2007, IASI, ROMANIA.

[14] Al-Rashedi, A.A., "E-Government Based on Cloud Computing and Service Oriented Architecture", *International Journal of Computer and Electrical Engineering*, Vol. 6, No. 3, 2014, pp. 201-206.

[15] Nahak, S.K, Mohapatra, D.P., Patra, M.R., "Survey on Service Oriented Architecture Testing", *International Journal of Advance Computing Techniques and Applications*, Vol. 2, No. 2, 2014, pp. 102-105.

[16] Belani, Y.R., Patel, S.S., "A Survey on Approach for Service Selection", *International Journal for Scientific Research & Development*, Vol. 3, No. 3, 2015, pp. 1499-1501.

[17] Depkominfo, "*Kerangka Acuan dan Pedoman Interoperabilitas Sistem Informasi Instansi Pemerintahan*", Jakarta: Direktorat Sistem Informasi, Perangkat Lunak dan Konten, Direktorat Jenderal Aplikasi Telematika, Departemen Komunikasi dan Informatika, 2008.

[18] Mumbaikar, S., Padiya, P., "Web Services Based On SOAP and REST Principles", *International Journal of Scientific and Research Publications*, Vol. 3, No. 5, 2013, pp. 1-4.

[19] Sirisha, K.L.S., Kumar, V.S., "Role of Web Services in Distributed Heterogeneous Applications and Testing for Quality Assurance", *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, Vol. 6, No. 6, 2017, pp. 100-103.

[20] Abeysinghe, S., "*RESTful PHP Web Services*". Birmingham-Mumbai: PACKT Publishing, 2008.

[21] Oladosu, J.B., Ajala, A.F., Popoola, O.O., "On The Use of Web Services Technology in E-Health Applications", *Journal of Theoretical and Applied Information Technology*, Vol. 12, No. 2, 2010, pp. 94-102.

[22] Utomo, W. H., "B2B Integration based on SOA Using Web Service", *International Journal of*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Computer Applications, Vol. 32, No. 2, 2011, pp. 41-48.

[23] Halili, F., Ramadani, E., "Web Services: A Comparison of Soap and Rest Services", *Modern Applied Science*, Vol. 12, No. 3, 2018, pp. 175-183.

[24] Warvante, M.S., "A Survey on Developing Web Services with SOAP and REST", *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, Vol. 6, No. 10, 2017, pp. 118-124.

[25] Kumari, V., "Web Services Protocol: SOAP vs REST", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, Vol. 4, No. 5, 2015, pp.2467-2469.

[26] Wagh, K., Thool, R., "A Comparative Study of SOAP Vs Rest Web Services Provisioning Techniques for Mobile Host", *Journal of Information Engineering and Applications*, Vol. 2, No. 5, 2012, pp. 12-16.

[27] Tihomirovs, J., Grabis, J., "Comparison of SOAP and REST Based Web Service Using Software Evaluation Metrics", *Information Technology and Management Science*, Vol. 19, 2016, pp. 92-97.

DESIGN RESTFUL WEB SERVICE OF NATIONAL POPULATION DATABASE FOR SUPPORTING E-HEALTH INTEROPERABILITY SERVICE

ORIGINALITY REPORT

97%

SIMILARITY INDEX

PRIMARY SOURCES

1	www.jatit.org Internet	5212 words — 91%
2	eprints.perbanas.ac.id Internet	336 words — 6%
3	jatit.org Internet	18 words — < 1%

EXCLUDE QUOTES	OFF	EXCLUDE MATCHES	OFF
EXCLUDE BIBLIOGRAPHY	OFF		