

Pengembangan PDPT Berbasis Framework Dengan Teknologi Web Service SOAP dan REST di Universitas Semarang

W. Adhiwibowo¹, G. G. Setiaji² dan T. J Kumkamdhani.³

^{1,2,3}Jurusan Teknik Informatika, Fakultas Teknologi Informasi dan Komunikasi, Universitas Semarang Gedung M Lantai 1, Jl. Soekarno Hatta, Tlogosari Kulon, Kec. Pedurungan, Kota Semarang
E-mail : whisnu@usm.ac.id¹, gallet@usm.ac.id², tirta@usm.ac.id³

Abstract— One indication of the development of a tertiary institution can be seen from the number of students applying from year to year. The increasing number of students is an identification that the tertiary institution is in high demand by the community. University student data is stored in a university database or often called PDPT. PDPT data is used for various things, such as calculating the ratio of lecturers and students, calculating the number of graduations, calculating class ratios to university income. Because of the importance of the PDPT data, the validity of the PDPT data must be maintained. The problem that arises at the University of Semarang is that the data is only contained in the central server and is not distributed well globally or in detail up to the Study Program and the Quality Assurance Agency. This requires an alternative way to validate student data so that student data can be monitored continuously. The method can be done by using software development methods using a framework. Data obtained through the computer center by utilizing a web service so that data can be retrieved directly.

Abstrak— Salah satu indikasi perkembangan suatu perguruan tinggi dapat terlihat dari jumlah mahasiswa yang mendaftar dari tahun ke tahun. Semakin meningkat jumlah mahasiswanya maka merupakan identifikasi bahwa perguruan tinggi tersebut sangat diminati oleh masyarakat. Data mahasiswa universitas tersimpan dalam pangkalan data perguruan tinggi atau sering disebut PDPT. Data PDPT digunakan untuk berbagai hal, seperti perhitungan rasio dosen dan mahasiswa, perhitungan jumlah kelulusan, perhitungan rasio kelas hingga pendapatan universitas. Karena pentingnya data PDPT tersebut, validitas dari data PDPT harus terus terjaga. Permasalahan yang muncul di Universitas Semarang adalah data hanya terdapat di dalam peladen pusat dan tidak terbagikan dengan baik secara global atau secara detail hingga ke Program Studi dan Badan Penjaminan Mutu. Untuk itu dibutuhkan cara alternatif untuk melakukan validasi dari data mahasiswa agar data mahasiswa dapat terpantau secara terus-menerus. Metode yang dapat dilakukan dengan cara menggunakan metode pengembangan perangkat lunak dengan memanfaatkan framework. Data diperoleh melalui puskom dengan memanfaatkan web service sehingga data dapat diambil secara langsung.

Kata Kunci—Web Service, SOAP, REST, PDPT

I. PENDAHULUAN

Salah satu indikasi perkembangan suatu perguruan tinggi dapat terlihat dari jumlah mahasiswa yang mendaftar dari tahun ke tahun. Semakin meningkat jumlah mahasiswanya maka merupakan identifikasi bahwa perguruan tinggi tersebut sangat diminati oleh masyarakat. Salah satu universitas yang diminati adalah Universitas Semarang dengan jumlah mahasiswa aktif mencapai 15.617 mahasiswa di Semester Gasal Tahun Akademik 2018/2019

Universitas Semarang yang berdiri sejak tahun 1987 dengan nama Politeknik Semarang, dan hanya memiliki 4 program studi. Saat ini Universitas Semarang memiliki 14 program studi yang aktif dengan 21.393 mahasiswa yang aktif. Jumlah mahasiswa yang cukup besar untuk suatu perguruan swasta di Jawa Tengah. Kondisi tersebut membutuhkan pengelolaan data yang baik dan valid, karena data tersebut harus dilaporkan ke pada PDDIKTI.

Data mahasiswa universitas tersimpan dalam pangkalan data perguruan tinggi atau sering disebut PDPT. Data PDPT digunakan untuk berbagai hal, seperti perhitungan rasio dosen dan mahasiswa, perhitungan jumlah kelulusan, perhitungan rasio kelas hingga pendapatan universitas. Karena pentingnya data PDPT tersebut, validitas dari data PDPT harus terus terjaga. Permasalahan yang muncul di Universitas Semarang adalah data hanya terdapat di dalam peladen pusat dan tidak terbagikan dengan baik secara global atau secara detail hingga ke Program Studi dan Badan Penjaminan Mutu. Sehingga jika diperlukan

pengecekan atau evaluasi terhadap mahasiswa tidak aktif ataupun aktif, maka data harus diminta secara manual ke Pusat Komputer dengan menggunakan *worksheet*.

Untuk itu dibutuhkan cara alternatif untuk melakukan validasi dari data mahasiswa agar data mahasiswa dapat terpantau secara terus-menerus. Metode yang dapat dilakukan dengan cara menggunakan metode pengembangan perangkat lunak dengan memanfaatkan framework. Data diperoleh melalui puskom dengan memanfaatkan web service sehingga data dapat diambil secara langsung.

I. METODE PENELITIAN

A. Penelitian Sebelumnya

Penelitian teknologi mengenai web service yang sebelumnya telah dilakukan menghasilkan sebuah sistem yang mampu mengintegrasikan sistem dan database dengan baik dengan menggunakan metode RESTFUL[1]. Yang di mana kesalahan pemasukan data mahasiswa sering terjadi jika dilakukan secara manual. Dalam implementasi metode REST juga dapat dilakukan dengan menggunakan framework lainnya sehingga dari penelitian yang sebelumnya dilakukan menghasilkan sebuah sistem dengan dua jenis modul API. Modul-modul API ini terformatkan JSON[2].

Dalam penggunaannya di sektor pendidikan, web service ini dapat membantu memonitor proses belajar mahasiswa sehingga mahasiswa yang memiliki kekurangan nilai untuk dapat berkonsultasi segera dengan dosen walinya. Penelitian ini menghasilkan sebuah sistem berbasis web dengan framework CodeIgniter yang memadukan sistem pangkalan data dengan sistem klien[3].

B. Tahap Pengumpulan Data

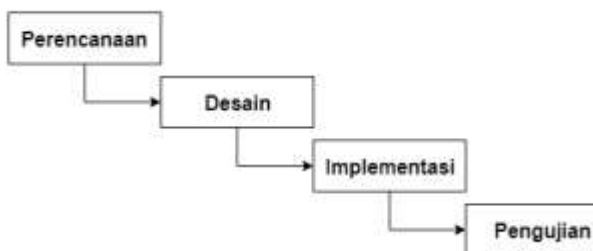
Untuk melakukan pengumpulan data, maka dalam penelitian ini akan menggunakan teknik wawancara kepada para *stackholder* baik yang memegang data tersebut maupun kepada pengguna yang menginginkan akses data tersebut. Hasil pengumpulan data inilah nantinya yang akan menjadi data primer dari sistem yang akan dikembangkan.

C. Dokumentasi System Requirement Specification

Data primer ini kemudian akan diproses menjadi sebuah dokumentasi yang berisikan segala kebutuhan-kebutuhan akan sistem yang diinginkan oleh *stackholder*. Dokumentasi kebutuhan ini bisa disebut dengan *Software Requirements Specification*[4] atau disebut juga dengan Spesifikasi Kebutuhan Perangkat Lunak. Dokumen SRS ini berisikan segala kebutuhan-kebutuhan fungsional dari sistem yang diinginkan berupa *query data, data categorization, print data*, dan lain-lain. Kebutuhan non-fungsional dari sistem ini dapat berupa kinerja sistem, keamanan sistem, bahkan tampilan antarmuka sistem[5]. Dokumen inilah yang akan menjadi acuan dari proses desain, implementasi atau pembangunan, pengujian, dan perilsan sistem.

D. Pengembangan Sistem

Dalam mengembangkan sistem yang akan dibuat terdapat beberapa tahap sesuai dengan *Software Development Life Cycle* menggunakan metode *Waterfall*. Metode *Waterfall* ini memiliki beberapa komponen utama seperti: Perencanaan; Desain; Implementasi; dan Pengujian. Tahapan *Waterfall* dimulai dan berakhir sebelum memulai selanjutnya dan berlaku secara linier[6].



Gambar. 1 Siklus Kehidupan Pengembangan Perangkat Lunak

1. Perencanaan

Di bagian ini sistem yang ingin dibangun dikumpulkan terlebih dahulu semua kebutuhan-kebutuhan yang diperlukan oleh *stackholder*. Semua informasi kebutuhan-kebutuhan ini kemudian didokumentasikan menjadi dokumen SRS yang mengandung kebutuhan fungsional dan non-fungsional.

2. Desain

Di bagian ini dokumentasi SRS tersebut kemudian menjadi dasar dari desain yang ingin dibangun. Desain ini dapat berupa cetak biru aplikasi, mockup, maupun sketsa

aplikasi. Peneliti dapat melakukan desain fungsionalitas sistem yang akan dibuat menggunakan *Unified Modeling Language*.

3. Implementasi

Di bagian ini desain yang telah dibuat baik desain fungsional maupun non-fungsional diimplementasikan menggunakan *framework* CodeIgniter dengan bahasa pemrograman PHP. Sistem dibangun dengan menggunakan dua jenis metode *web service* yaitu dengan metode SOAP dan REST.

4. Pengujian

Di bagian ini sistem yang telah selesai dibangun menggunakan bahasa pemrograman dan *framework* CodeIgniter diuji sesuai dengan dokumentasi SRS. Implementasi fungsional diuji menggunakan teknik *Blackbox* dan non-fungsional diuji dengan menggunakan perbandingan kinerja.

II. HASIL DAN PEMBAHASAN

A. Persiapan dan Perencanaan

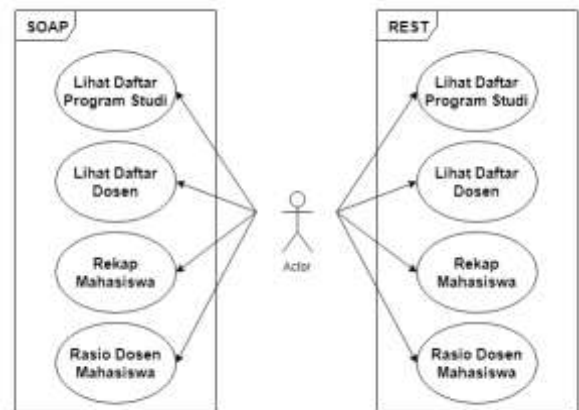
Data yang digunakan untuk penelitian ini didapatkan dari Forlap Dikti melalui Feeder. Data yang digunakan untuk meneliti *web service* SOAP dan REST yaitu data aktivitas mahasiswa sebanyak 448.815 record. Kemudian data dinormalisasikan, yaitu data yang lebih spesifik dengan acuan semester Genap 2018/2019 didapatkan data sebanyak 21.463. Formula Matematika.

Perlunya normalisasi ini dilakukan karena untuk melihat atau meneliti apakah dengan data sebanyak ini dapat ditampilkan secara maksimal atau tidaknya.

B. Desain Sistem

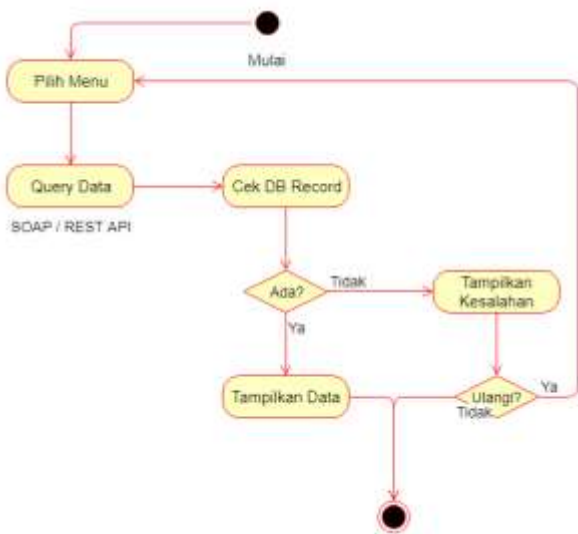
Sistem yang akan dibangun harus didesain terlebih dahulu untuk menghindari kesalahan ketika mengimplementasikan fungsi-fungsi ke dalam sistem yang dibuat.

Dalam mendesain sebuah sistem, hal pertama kali yang bisa di desain adalah skenario bagaimana pengguna menggunakan sistem tersebut. Skenario-skenario ini diharuskan berdasar kepada dokumentasi SRS yang telah dilakukan, maka skenario yang terbentuk adalah Lihat Daftar Program Studi; Lihat Daftar Dosen; Rekap Mahasiswa; dan Rasio Dosen Mahasiswa.



Gambar. 2 Desain Use Case Diagram dari Sistem

Di bagian ini akan digambarkan bagaimana aktivitas yang berjalan di dalam sistem yang akan dibuat. Antara SOAP, REST dan fungsi-fungsi di dalamnya memiliki alur umum yang sama dalam hal pemrosesan *query data*. Dikarenakan karakteristik aktivitas yang sama, maka keseluruhan fungsi yang ada disingkat sesuai dengan Gambar 3.



Gambar. 3 Desain Activity Diagram dari Sistem

C. Implementasi

Dari desain yang telah dibuat akan diimplementasikan menggunakan framework CodeIgniter dengan bahasa pemrograman PHP. Namun dalam implementasinya akan menggunakan dua metode yang berbeda yaitu metode *Simple Object Access Protocol* dan *Representation State Transfer*.

1. Metode SOAP yaitu sebuah mekanisme dari *Remote Procedure Call* yang di mana interaksi antara peladen web dan klien menggunakan dokumentasi XML sebagai pemicu pesan permintaan (*request*) dan jawaban (*response*), sertamenggunakan protokol HTTP sebagai media transportasi data [7]. Dengan menggunakan metode ini, data mahasiswa akan dikirimkan menggunakan XML dan melalui medium HTTP.

2. Metode REST API adalah sebuah metode alternatif dari SOAP yang lebih simpel dengan menggunakan *Uniform Resource Identifier* (URI). Sehingga metode ini lebih populer dibandingkan dengan generasi sebelumnya [8]. Dalam implementasinya nanti *web Service* akan dibangun menjadi dua yaitu bagian peladen dan klien.

D. Pengujian

Metode-metode yang telah diimplementasikan kemudian diuji dengan cara perbandingan kinerja ketika melakukan *query data* ke peladen. Dalam pengujian setiap metode akan dmelakukan 2 jenis *query* dengan jumlah *field* yang berbeda. Dari sinilah indikator kinerja metode yang diimplementasikan dapat diukur secara metrik.

1. Hasil pengujian kinerja metode SOAP

Metode ini diuji dengan mengambil data dengan 1 dan 15 *field* dengan menggunakan peramban *Google Chrome*. Indikator yang digunakan sebagai pengukur kinerja dari metode adalah Jumlah Field Tabel, Jumlah Record, Estimasi Waktu yang Diperlukan, dan Ukuran Data yang

Terambil. Dari proses pengambilan data itulah dihasilkan data sebagai berikut:

Tabel 1.

Hasil Pengujian Menggunakan Metode SOAP

	SOAP	
Field Table	1	15
Jumlah Record	97,010	15,116
Estimasi Waktu	8.21s	4.97s
Ukuran	2.1 MB	505 KB

2. Hasil pengujian kinerja metode REST

Metode REST juga akan mengalami proses pengujian dengan indkator yang sama dengan sebelumnya. Sehingga dari proses pengujian tersebut dihasilkan data sebagai berikut:

Tabel 2.

Hasil Pengujian Menggunakan metode REST

	REST	
Field Table	1	15
Jumlah Record	161,092	25,577
Estimasi Waktu	15.91s	9.17s
Ukuran	4.1MB	2.2MB

Dari pengujian yang telah dilakukan bisa sedikit disimpulkan bahwa metode REST dapat melakukan pengambilan lebih banyak dibandingkan dengan metode SOAP.

III. KESIMPULAN

Dari penelitian diatas tentang metode webservice SOAP dan REST API dapat ditarik kesimpulan sebagai berikut :

1. Metode webservice REST lebih unggul dari SOAP karena dalam prosesnya REST tidak memerlukan bandwidth yang banyak sehingga aksesnya lebih cepat.
2. Webservice REST API dapat menampilkan data maximal 16,1092 record untuk 1 field data, sedangkan untuk 15 field didapatkan 25,577 record.
3. Sedangkan SOAP hanya dapat menampilkan data maximal 97,010 record untuk 1 field data, sedangkan 15 field didapatkan data sebanyak 15,116 record.

Penelitian tentunya memiliki beberapa kekurangan diberbagai aspek. Sehingga kedepannya ada beberapa saran yang dapat menjadi inspirasi untuk penelitian berikutnya berupa: penggunaan perangkat keras yang berbeda, DBMS yang berbeda, framework yang berbeda, atau menggunakan metode *web service* lainnya.

DAFTAR PUSTAKA

- [1] T. Andriyanto and R. Indriati, "RESTful Web Service Untuk Integrasi Data Antar Sistem Informasi," *J. Sains dan Inform.*, 2017.
- [2] M. Agus, A. Teknik, S. T. T. Terpadu, and N. Fikri, "Analisis dan Perancangan Representational State Transfer (REST) Web Service Sistem Informasi Akademik STT Terpadu Nurul Fikri Menggunakan YII Framework," *J. Teknol. Terpadu*, 2016.
- [3] R. Somya, S. Supriyadi, and L. B. Prasetyo, "SISTEM MONITORING NILAI PROSES BELAJAR MENGAJAR PADA PROGRAM STUDI MENGGUNAKAN WEB SERVICE," *Din. Rekayasa*, 2018.
- [4] J. Kuchta, "Completeness and Consistency of the System Requirement Specification," in *Position Papers of the 2016 Federated Conference on Computer Science and Information Systems*, 2016.
- [5] S. Requirements, "IEEE Guide to Software Requirements Specifications," *East*, 1984.
- [6] A. Adel and B. Abdullah, "A Comparison Between Three SDLC Models Waterfall Model, Spiral Model, and Incremental/Iterative Model," *IJCSI Int. J. Comput. Sci. Issues*, 2015.
- [7] "SOAP: Simple object access protocol," *Electronic Design*. 2001.
- [8] A. Rodriguez, "Restful web services: The basics," *Online Artic. IBM Dev. Tech. Libr.*, 2008.
- [9] G. Sig and P. Masjid, "RANCANG BANGUN APLIKASI SISTEM INFORMASI," vol. 17, no. 1, pp. 34–40, 2019.
- [10] A. Tri, J. Harjanta, and B. A. Herlambang, "Rancang Bangun Game Edukasi Pemilihan Gubernur Jateng Berbasis Android Dengan Model ADDIE," vol. 16, no. 1, pp. 91–97, 2018.