

Web Services Menggunakan Format JSON

Robby Cokro Buwono

Sistem Informasi STMIK AKAKOM

Jl. Raya Janti Karang Jambe No. 143, Yogyakarta 55198 INDONESIA

robbycokro@akakom.ac.id

INTISARI

Sistem informasi dibuat untuk membantu kegiatan di suatu organisasi dan pengambilan keputusan para pimpinannya. Sistem informasi sudah banyak dibuat organisasi namun data atau informasi yang dihasilkan selalu berdiri sendiri sehingga mengakibatkan kesulitan mendapatkan informasi yang terkait antar sistem informasi pada organisasi tersebut, terjadi duplikasi informasi dan menjadikan informasi tidak berupa kesatuan yang lengkap dan utuh.

Web sebagai sarana penyampaian informasi yang sederhana dari satu tempat ke tempat lain melalui jaringan komputer. Web dapat dimanfaatkan pula sebagai sarana interoperabilitas dan berbagi informasi antara sistem informasi di suatu organisasi. Suatu klien sistem informasi yang berada pada jaringan komputer dapat meminta layanan data berupa web services kepada server sistem informasi yang menyediakan layanan data yang diminta dengan cara mudah.

JavaScript Object Notation (JSON) merupakan seperangkat aturan untuk memformat data berbasis teks yang ringan digunakan pada pertukaran data. Layanan web (web services) pada komunikasi antar sistem informasi untuk bertukar informasi dapat menggunakan JSON agar mendapatkan kemudahan mendapatkan informasi yang dibutuhkan dan untuk mendapatkan integritas data yang dihasilkan dari berbagai sistem informasi pada organisasi tersebut.

Peneliti melakukan pengembangan web services untuk berbagi data antar suatu sistem informasi yang dibutuhkan oleh sistem informasi yang lainnya dengan format menggunakan JSON. Peneliti mengharapkan adanya web services dengan format JSON yang dapat membantu sistem informasi untuk mendapatkan informasi yang dibutuhkan dan mendapatkan integritas data yang dihasilkan antara sistem informasi tersebut.

Kata kunci— *web services, JSON, interoperabilitas, sistem informasi.*

ABSTRACT

Information systems are made to assist activities in an organization and the decisions of its leaders. Many information systems have been created by the organization, but the data or information produced always stands alone, resulting in difficulties in getting information related to information systems in the organization, duplication of information and making the information, not in the form of a complete and intact unit.

The web as a means of delivering simple information from one place to another through computer networks. The web can also be used as a means of interoperability and information sharing between information systems in an organization. An information system client located on a computer network can request data services in the form of web services to an information system server that provides requested data services in an easy way.

JavaScript Object Notation (JSON) is a set of rules for formatting lightweight text-based data used in data exchange. Web services (web services) on communication between information systems to exchange information can use JSON to get the ease of getting the information needed and to obtain data integrity generated from various information systems in the organization.

The researcher develops web services to share data between information systems needed by other information systems in a format using JSON. The researcher expects the existence of web services with JSON format that can help information systems to get the information needed and get the integrity of the data generated between the information systems.

Kata kunci— *web services, JSON, interoperability, information systems.*

I. PENDAHULUAN

Suatu organisasi seperti perusahaan yang berskala besar akan sangat dibantu dalam kegiataannya dengan adanya penggunaan sistem informasi. Sistem informasi pada suatu organisasi mengatur pengelolaan informasi

dan menghasilkan informasi yang sangat bermanfaat bagi pengambilan keputusan di organisasi tersebut. Pada era sekarang ini banyak organisasi yang sudah sangat bergantung dengan sistem informasi bahkan sudah mengembangkan berbagai macam

sistem informasi untuk membantu kegiatan di organisasi tersebut. Namun, sistem informasi pada organisasi tersebut menghasilkan data atau informasi yang selalu berdiri sendiri sehingga mengakibatkan kesulitan mendapatkan informasi yang terkait antar sistem informasi pada organisasi tersebut, terjadi duplikasi informasi dan menjadikan informasi tidak berupa kesatuan yang lengkap dan utuh.

Web sebagai sarana penyampaian informasi yang sederhana dari satu tempat ke tempat lain melalui jaringan komputer. Web berjalan pada protokol HTTP (*Hypertext Transfer Protocol*), yaitu protokol yang mengatur transformasi informasi HTML (dokumen web) antara *server* dan klien. Penggunaan web tidak dibatasi oleh platform tertentu, dan tidak memerlukan pengaturan yang rumit antara komputer *server* dan klien sehingga memudahkan informasi dapat diakses dari berbagai tempat melalui jaringan komputer.

Web dapat dimanfaatkan sebagai sarana berbagi informasi antara sistem informasi. Suatu klien sistem informasi yang berada pada jaringan komputer dapat meminta layanan data berupa *web services* kepada *server* sistem informasi yang menyediakan layanan data yang diminta dengan cara mudah. Data dari *server* yang dilewatkan melalui web dengan fitur *web services* tersebut dibuat dengan format JSON untuk dikonversi menjadi data yang dibutuhkan oleh klien dan untuk mendapatkan integritas data yang dihasilkan dari berbagai sistem informasi pada organisasi tersebut. *JavaScript Object Notation (JSON)* merupakan seperangkat aturan untuk memformat data berbasis teks yang ringan digunakan pada pertukaran data.

Berdasarkan penjelasan di atas maka perumusan masalah penelitian ini adalah bagaimana mengembangkan layanan web (*web services*) yang digunakan untuk mentransformasi informasi dari satu sistem informasi ke sistem informasi yang lain dengan menggunakan format JSON. Tujuan dari penelitian ini adalah terciptanya layanan web (*web services*) yang dapat digunakan untuk berbagi data/informasi antar sistem informasi berbasis web dengan format JSON. Manfaat yang diharapkan dari penelitian ini adalah hasil pengembangan *web services* dengan format JSON untuk pertukaran informasi antar sistem informasi dapat diterapkan untuk organisasi yang memiliki dan mengembangkan sistem informasi.

II. METODOLOGI PENELITIAN

Tahapan penelitian Pengembangan *Web Services* Menggunakan Format JSON adalah sebagai berikut:

1. melakukan studi pustaka mengenai web, sistem informasi, interoperabilitas, *web services*, JSON dan rekayasa perangkat lunak untuk mendapatkan gambaran arsitektur *web services* yang akan dikembangkan;
2. melakukan pengembangan *web services* dengan format JSON pada *server* untuk membuat informasi dan pada klien untuk mendapatkan informasi yang diinginkan;
3. menguji kesesuaian dan kelengkapan data sebelum dan sesudah diformat dengan JSON pada penerapan *web services* untuk berbagi data mahasiswa antara sistem informasi penerimaan mahasiswa dan sistem informasi akademik di suatu Perguruan Tinggi.

Teknik pengumpulan data pada penelitian ini adalah:

1. Observasi dengan melakukan pengamatan langsung dengan objek yang diteliti yaitu pada pengembangan *web services*, pengujian dan penerapan *web services* untuk berbagi data antara sistem informasi;
2. Studi Pustaka tentang web, sistem informasi, *web services*, JSON dan rekayasa perangkat lunak untuk mendapatkan pemahaman tentang pembentukan arsitektur *web services* dan pengembangan *web services*.

Alat-alat yang digunakan pada penelitian ini adalah:

1. seperangkat komputer yang digunakan untuk pengembangan *web services*;
2. aplikasi editor pengolah HTML, CSS, Javascript dan PHP Script untuk pengembangan *web services*;
3. aplikasi *web server* dan *browser* untuk menjalankan sistem informasi.

Bahan yang digunakan pada penelitian ini adalah pustaka tentang *web services* dan data mahasiswa yang ditransformasi antara sistem informasi penerimaan mahasiswa baru dan sistem informasi akademik di suatu Perguruan Tinggi.

III. HASIL DAN PEMBAHASAN

Sistem informasi merupakan kumpulan elemen-elemen yang bekerja sama untuk mengolah data atau informasi yang berguna untuk pengambilan keputusan di organisasi. Web dapat dikatakan sebagai sistem informasi yang penyajiannya berupa hiperteks. Web

sebagai sarana untuk mendapatkan informasi dari *server* web kepada klien melalui jaringan komputer melalui protokol *HTTP* (*Hypertext Transfer Protocol*).

Web services merupakan suatu komponen *software* yang merupakan *self-containing*, aplikasi *modular self-describing* yang dapat dipublikasikan, dialokasikan, dan dilaksanakan pada web. *Web services* adalah teknologi yang mengubah kemampuan internet dengan menambahkan kemampuan *transactional web*, yaitu kemampuan web untuk saling berkomunikasi dengan pola *program-to-program* (*P2P*). Fokus web selama ini didominasi oleh komunikasi *program-to-user* dengan interaksi *business-to-consumer* (*B2C*), sedangkan *transactional web* akan didominasi oleh *program-to-program* dengan interaksi *business-to-business*[6].

Web services secara teknis memiliki mekanisme interaksi antar sistem sebagai penunjang interoperabilitas, baik berupa agregasi (pengumpulan) maupun sindikasi (penyatuan). *Web services* memiliki layanan terbuka untuk kepentingan integrasi data dan kolaborasi informasi yang bisa diakses melalui internet oleh berbagai pihak menggunakan teknologi yang dimiliki oleh masing-masing pengguna. Kelebihan *web services* adalah lintas platform, *language independent*, jembatan penghubung dengan *database* tanpa perlu *driver database* dan tidak harus mengetahui jenis *DBMS*, mempermudah proses pertukaran data, serta penggunaan kembali komponen aplikasi[5].

Interoperabilitas menurut IEEE, yaitu "*the ability of two or more systems or components to exchange information and use the exchanged information*". Dua persyaratan yang termasuk dalam definisi IEEE adalah keberadaan pertukaran dan persyaratan nyata bahwa siapa yang mendapatkan apa yang dipertukarkan menggunakannya untuk melakukan sesuatu merupakan aspek-aspek yang disorot dalam sebagian besar definisi, dengan demikian dianggap sebagai dua ide penting yang mendasari konsep interoperabilitas. Definisi interoperabilitas lainnya adalah menambah kemampuan untuk bertukar dan menggunakan data, kemampuan satu entitas untuk menggunakan fungsionalitas entitas lain. Selanjutnya situasi interoperabilitas dicirikan oleh gagasan "bertindak atas permintaan", yaitu satu entitas melakukan sesuatu sebagai tanggapan terhadap permintaan yang diterima dari entitas lain (entitas peminta). Dengan demikian, hanya ada interoperabilitas antara dua entitas

A dan B jika entitas A mampu mengirim permintaannya ke entitas B dan entitas B dapat menerima permintaan itu, untuk memahaminya dan melakukan sesuatu yang benar-benar sesuai dengan tindakan yang entitas A dimaksudkan untuk melihat dieksekusi oleh entitas B dalam menanggapi permintaan yang dibuat [3].

Web services digunakan sebagai suatu fasilitas yang menyediakan layanan (dalam bentuk informasi atau data) kepada sistem lain, sehingga dapat berinteraksi dengan sistem tersebut melalui layanan-layanan yang disediakan. *Web services* menyimpan data informasi dalam format JSON atau XML, sehingga data ini dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi, dan bahasa pemrograman[7].

JSON adalah sintaksis ringan, berbasis teks, dan tidak tergantung pada suatu bahasa tertentu untuk menentukan format pertukaran data. JSON berasal dari bahasa pemrograman ECMAScript, merupakan bahasa pemrograman yang independen. JSON mendefinisikan seperangkat kecil aturan untuk mengatur representasi secara portabel dari data terstruktur[9].

JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh *programmer* keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data. JSON terbuat dari dua struktur:

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar kunci (*keyed list*), atau associative array.
2. Daftar nilai yang terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*) atau urutan (*sequence*).

Struktur tersebut adalah struktur data universal, hampir semua bahasa pemrograman modern mendukung struktur tersebut dalam satu bentuk atau bentuk yang lain. Dengan demikian bahwa format data bisa dipertukarkan dengan bahasa pemrograman didasarkan pada struktur tersebut[11].

Referensi [1] memberikan simpulan bahwa penggunaan JSON sebagai media pertukaran data yang ringan dan mudah diterjemahkan oleh komputer maupun manusia dikarenakan kombinasi kode yang membentuk *array*

mudah dibuat dan dipahami. JSON itu sendiri sangat berguna untuk mendukung perancangan aplikasi pada perguruan tinggi dengan menggunakan metode *REST full API* dan *web services*.

Referensi [2] memberikan kesimpulan bahwa *web services* dengan format data JSON dapat dipergunakan untuk mengintegrasikan data pada *client* yaitu data kategori matakuliah dan data nilai pada *server*. Data dari *web response* dapat diolah lebih lanjut menjadi data potensi yang terurut dari yang paling berpotensi. Pemanggilan *web services* dilakukan dengan menggunakan function curl pada PHP.

Referensi [10] memberikan kesimpulan bahwa pengimplementasian *web services* dan Android pada *Digital Library Application* Politeknik Sukabumi menggunakan JSON dapat dijadikan solusi agar sistem informasi perpustakaan dapat berjalan lebih baik lagi.

Referensi [8] mengkaji bahwa prototipe sistem yang dikembangkan dapat memberikan informasi yang terintegrasi melalui sebuah aplikasi berbasis web yang dikembangkan dengan menggunakan PHP sebagai klien dari dua *web services* yang ada. Implementasi teknologi *web services* yang berfungsi sebagai middleware mampu melakukan pertukaran pesan (*message*) dengan memanfaatkan protocol HTTP melalui sebuah jaringan antara aplikasi dan *database* antara rumah sakit pemberi rujukan (puskesmas) ke rumah sakit rujukan (AMC/RSU).

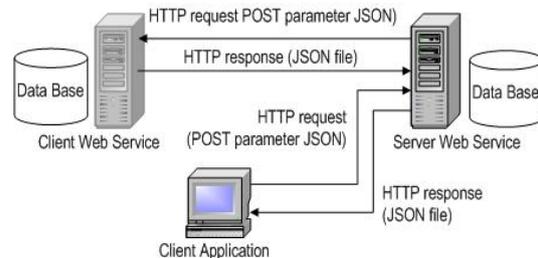
Aplikasi mengirim request pemesanan ke perusahaan melalui *application server* perusahaan dan disimpan di basis data. *Request* dari aplikasi *mobile* akan dikirim ke *application server* dengan teknologi *web services* dalam bentuk *JavaScript Object Notation (JSON)* menggunakan layanan Internet. *Web services* merupakan sistem perangkat lunak yang terdiri atas kumpulan fungsi atau *methods* yang tersimpan pada sebuah *server* dan dapat dipanggil oleh klien. Dengan *web services* layanan dan aplikasi dapat digunakan ulang, dan memungkinkan terjadinya pertukaran data antar platform yang berbeda. Teknologi *web services* telah dimanfaatkan dalam bidang kesehatan, bisnis, integrasi data di kepolisian, *e-government* dan sebagainya. Antarmuka *web services* dideskripsikan menggunakan format yang mampu diproses oleh mesin[4].

Melihat tinjauan pustaka dan kajian-kajian penelitian sebelumnya peneliti mendapatkan penjelasan tentang *web services*, protokol yang digunakan serta penggunaan JSON pada

web services. Hal-hal tersebut menjadi bahan atau rujukan penelitian untuk menggambarkan arsitektur *web services* dengan format JSON dan mengembangkan *web services* pada *server* penyedia layanan dan klien yang menggunakan *web services* dengan format JSON. Menerapkan *web services* yang dikembangkan untuk transformasi data mahasiswa antara sistem informasi penerimaan mahasiswa dan sistem informasi akademik.

A. Arsitektur Web Services

Web services adalah sarana komunikasi data antara mesin dengan mesin lainnya dalam jaringan komputer dengan memanfaatkan media protokol HTTP atau web. Arsitektur *web services* pada jaringan aplikasi / sistem informasi adalah menggunakan *multi-tier* seperti ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur Web Services.

B. Analisis Kebutuhan Sistem

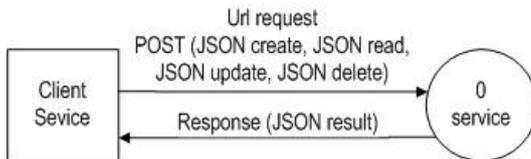
Kebutuhan sistem pada *web services* dengan format JSON adalah sebagai berikut:

1. Sesuai dengan gambar Arsitektur *Web Services* seperti Gambar 1, sistem *web services* dikembangkan di tiga mesin yaitu *Server Web Services*, *Client Web Services* dan *Client Application*. *Server Web Services* dan *Client Web Services* adalah *server* web dengan sistem informasi yang memiliki akses ke basisdata-nya masing-masing. *Client Application* adalah klien atau pengguna akhir yang akses ke *Server Web Services* melalui web.
2. *Server Web Services* menyediakan layanan untuk pengelolaan basisdata meliputi operasi tambah, ubah, hapus dan baca data (*insert, update, delete, select*) pada suatu tabel dalam basisdata yang dimilikinya. Layanan pada *Server Web Services* diakses melalui web dengan menerima parameter dan menghasilkan layanan dalam format JSON.

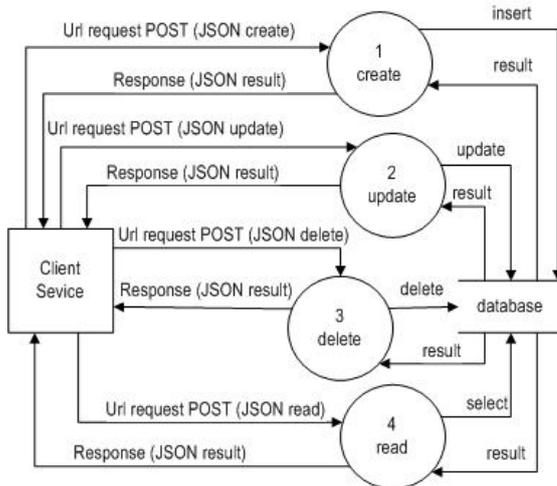
3. Client Web Service dan Client Application dapat menggunakan layanan yang disediakan oleh Server Web Services.
4. Client Web Services dapat meneruskan operasi tambah, ubah, hapus dan baca data (*insert, update, delete, select*) pada suatu tabel dalam basisdata yang dimilikinya sesuai dengan kebutuhan pada sistem informasi.

C. Perancangan Sistem

Perancangan sistem adalah model yang menjelaskan secara logika tentang sistem informasi yaitu *web services* yang dikembangkan ini saat nanti ketika bekerja, ditunjukkan oleh Gambar 2 - 3.



Gambar 2. Diagram Konteks Web Services.



Gambar 3. Diagram Level 1 Web Services.

D. Implementasi Sistem

Penulisan kode program dibagi menjadi tiga mesin yaitu pada *Server Web Services, Client Web Services* dan *Client Application*.

Berikut adalah cuplikan kode program dari algoritma *web service* yang dikembangkan:

```
1. Menyisipkan data di Server Web Services adalah
public function insert($insert)
{
    // $insert =
    ["_t"=>"tableName", "_s"=>["fieldName"=>"value", ...
]]
```

```
if(!isset($insert["_t"]) || empty($insert["_t"]))
    return false ;
if(!isset($insert["_s"]) || count($insert["_s"]) <=0)
    return false ;
$sqlInsert = "" ;
$sqlInsert .= " insert into ".$insert["_t"]." (" ;
$i=0 ;
foreach($insert["_s"] as $fieldName => $value){
    if($i != 0)
        $sqlInsert .= ", " ;
    $sqlInsert .= $fieldName ;
    $i++ ;
}
$sqlInsert .= ") values (" ;
$i=0 ;
foreach($insert["_s"] as $fieldName => $value){
    if($i != 0)
        $sqlInsert .= ", " ;
    $sqlInsert .= "".$value."" ;
    $i++ ;
}
$sqlInsert .= ") " ;
return $sqlInsert ;
// insert into tableName ( fieldName_1,
fieldName_2 ) values ( 'value_1', 'value_2' )
}
```

2. Membaca data di *Server Web Services* adalah

```
public function select($select)
{
    // $select = ["_t"=>"tableName",
    "_f"=>["fieldName",...], "_c"=>["fieldNameCondition"
=>"valueCondition",...], "_l"=>"offset,limit / limit"]
    if(!isset($select["_t"]) || empty($select["_t"]))
        return false ;
    $sqlSelect = "" ;
    $sqlSelect .= "select " ;
    if(isset($select["_f"]) && is_array($select["_f"]) &&
count($select["_f"])>0){
        $i=0 ;
        foreach($select["_f"] as $fieldName ){
            if($i != 0)
                $sqlSelect .= ", " ;
            $sqlSelect .= $fieldName ;
            $i++ ;
        }
    }else{
        $sqlSelect .= " * " ;
    }
    $sqlSelect .= " from ".$select["_t"] ;
    if(isset($select["_c"]) && count($select["_c"])>0){
        $i = 0 ;
        foreach($select["_c"] as $fieldName => $value){
            if($i == 0)
                $sqlSelect .= " where ".$fieldName ;
            else
                $sqlSelect .= " and ".$fieldName ;
        }
    }
}
```

```

    if(is_numeric($value))
        $sqlSelect .= " = '" . trim($value) . "'";
    else
        $sqlSelect .= " like '%" . $value . "%' ";
    $i++;
}
}
if(isset($select["_l"]) && !empty($select["_l"])){
    $l = explode(",",$select["_l"]);
    if(isset($l[0]) and isset($l[1])){
        $offset = (int) $l[0];
        $range = (int) $l[1];
        $sqlSelect .= " limit " . $offset . ", " . $range;
    }elseif(is_numeric($select["_l"])){
        $sqlSelect .= " limit " . $select["_l"];
    }else{
        $sqlSelect .= "";
    }
}
}
}
}
return $sqlSelect;
// select fieldName_1, fieldName_2 from
tableName where fieldNameCondition like
'valueCondition%' limit 999,999
}

```

3. Memperbarui data di *Server Web Services* adalah

```

public function update($update)
{
    // $update =
    ["_t"=>"tableName", "_s"=>["fieldName"=>"value",...
    ], "_c"=>["fieldNameCondition"=>"valueCondition",...
    ]]
    if(!isset($update["_t"]) || empty($update["_t"]))
        return false;
    if(!isset($update["_s"]) || count($update["_s"])<=0)
        return false;
    if(!isset($update["_c"]) || count($update["_c"])<=0)
        return false;
    $sqlUpdate = "";
    $sqlUpdate .= "update " . $update["_t"];
    $sqlUpdate .= " set ";
    $i=0;
    foreach($update["_s"] as $fieldName => $value){
        if($i != 0)
            $sqlUpdate .= ", ";
        $sqlUpdate .= $fieldName . " = '" . $value . "'";
        $i++;
    }
    $i = 0;
    foreach($update["_c"] as $fieldName => $value){
        if($i == 0)
            $sqlUpdate .= " where " . $fieldName;
        else
            $sqlUpdate .= " and " . $fieldName;
        $sqlUpdate .= " = '" . $value . "'";
    }
}

```

```

    $i++;
}
}
return $sqlUpdate;
// update tableName set fieldName_1 = 'value_1',
fieldName_2 = 'value_2' where fieldNameCondition
= 'valueCondition'
}
}

```

4. Menghapus data di *Server Web Services* adalah

```

public function delete($delete)
{
    // $delete =
    ["_t"=>"tableName", "_c"=>["fieldNameCondition"=>
    "valueCondition",...]]
    if(!isset($delete["_t"]) || empty($delete["_t"]))
        return false;
    if(!isset($delete["_c"]) || count($delete["_c"])<=0)
        return false;
    $sqlDelete = "";
    $sqlDelete .= "delete from " . $delete["_t"];
    $i = 0;
    foreach($delete["_c"] as $fieldName => $value){
        if($i == 0)
            $sqlDelete .= " where " . $fieldName;
        else
            $sqlDelete .= " and " . $fieldName;
        $sqlDelete .= " = '" . $value . "'";
        $i++;
    }
}
return $sqlDelete;
// delete from tableName where
fieldNameCondition = 'valueCondition'
}

```

5. Memberikan tanggapan hasil ke klien di *Server Web Services* adalah

```

public function responseJson($jsonInput)
{
    $this->jsonOutput = '{"_r":false}';
    $result = false;
    $this->jsonInput = $jsonInput;
    $valid = $this->validateJsonInput();
    if($valid){
        $arrData = $this->arrData;
        if(isset($arrData["_c"])){
            $this->arrSql = $arrData["_c"];
            $this->txtResponseJson = "create";
            $result = $this->create();
        }elseif(isset($arrData["_r"])){
            $this->arrSql = $arrData["_r"];
            $this->txtResponseJson = "read";
            $result = $this->read();
        }elseif(isset($arrData["_u"])){
            $this->arrSql = $arrData["_u"];
            $this->txtResponseJson = "update";
            $result = $this->update();
        }elseif(isset($arrData["_d"])){
            $this->arrSql = $arrData["_d"];
        }
    }
}

```

```

    $this->txtResponseJson = "delete" ;
    $result = $this->delete() ;
  }else{
    $this->jsonOutput = '{"_r":false}';
    $this->txtResponseJson = "not-thing" ;
    $result = false ;
  }
}
return $result ;
}

```

6. Mengirimkan permintaan ke *server* di *Client Web Services* adalah

```

public function sendRequest()
{
  if(!$this->validateJsonInput()){
    return false ;
  }
  $post = [
    '_req' => $this->jsonInput ,
  ];
  $ch = curl_init($this->url);
  curl_setopt($ch, CURLOPT_RETURNTRANSFER,
  true);
  curl_setopt($ch, CURLOPT_POSTFIELDS, $post);
  $response = curl_exec($ch);
  $httpCode = curl_getinfo($ch,
  CURLINFO_HTTP_CODE);
  if ($httpCode == 404) {
    $this->txtResult .= " server service tidak
    ditemukan (404) \n" ;
  }
  curl_close($ch);
  $this->jsonOutput = trim($response) ;
  $validJsonOutput = $this->validateJsonOutput() ;
  if($validJsonOutput){
    return true ;
  }else{
    return false ;
  }
}

```

7. Mengirimkan permintaan ke *server* di *Client Web Services* adalah

```

<script>
$(document).ready(function(){
  $("#request-data").click(function(){
    var _dataObjek = {
      _r:{
        _t:'mahasiswa',
        _f:["no_pendaftaran","nama_mahasiswa","alamat_
        mahasiswa","tgl_daftar","angkatan","kode_prodi","t
        anggal_lahir","tempat_lahir","email","no_telp","aga
        ma","jenis_kelamin","nama_sma","tahun_tamat_sm
        a","no_ijazah_sma","tanggal_ijazah_sma","nilai_u
        n_sma"],
        _c: { no_pendaftaran:'1800028' },
        _l:"

```

```

    }
  });
  var _data = JSON.stringify(_dataObjek) ;
  var url = "http://localhost/webservice/webservice-
  2018/appserver/service.php" ;
  $.post(url, { _req : _data },
  function(result){
    var element = document.createElement('a');
    element.setAttribute('href',
    'data:application/json;charset=utf-8,' +
    encodeURIComponent(result));
    // element.setAttribute('download', 'result.json');
    element.style.display = 'none';
    document.body.appendChild(element);
    element.click();
    document.body.removeChild(element);
  },
  "text");
})
}
</script>

```

Kode program yang telah dibuat selanjutnya dilakukan pengujian dengan mencoba sistem informasi untuk bekerja. Pengujian sistem informasi tersebut bertujuan untuk mendapatkan sistem informasi yang dapat bekerja dengan baik sesuai dengan analisis dan perancangan sistem yang telah dibuat sebelumnya. Hasil pengujian terakhir didapatkan bahwa *web services* sudah dapat bekerja sesuai dengan analisis kebutuhan ditunjukkan seperti ditunjukkan pada Tabel I.

TABEL I.
HASIL AKHIR PENGUJIAN WEB SERVICES

No.	Fungsi Sistem	Hasil Uji
1.	<i>Web services</i> berhasil dikembangkan mesin <i>Server Web Services</i>	Ok
2.	<i>Web services</i> berhasil dikembangkan mesin <i>Client Web Services</i>	Ok
3.	<i>Web services</i> berhasil dikembangkan mesin <i>Client Application</i>	Ok
4.	<i>Server Web Services</i> menyediakan layanan untuk pengelolaan basisdata meliputi operasi tambah, ubah, hapus dan baca data diakses melalui web dengan menerima parameter dan menghasil layanan dalam format JSON.	Ok
5.	<i>Client Web Service</i> dapat menggunakan layanan yang disediakan oleh <i>Server Web Services</i> .	Ok
6.	<i>Client Web Services</i> dapat meneruskan operasi tambah, ubah, hapus dan baca data (<i>insert, update, delete, select</i>) pada suatu tabel dalam basisdata yang dimilikinya.	Ok
7.	<i>Client Application</i> dapat menggunakan layanan yang disediakan oleh <i>Server Web Services</i> .	Ok

Web services yang sudah dikembangkan selanjutnya diterapkan untuk berbagi data mahasiswa antara sistem informasi penerimaan mahasiswa dan sistem informasi akademik di suatu Perguruan Tinggi. Penerapan *web services* dimulai dengan melakukan pemetaan *field* pada sistem informasi penerimaan mahasiswa dengan sistem informasi akademik ditunjukkan pada Tabel II.

TABEL II.

PEMETAAN *FIELD* DATA MAHASISWA DI SISTEM INFORMASI AKADEMIK DENGAN SISTEM INFORMASI PENERIMAAN MAHASISWA

<i>Field data calon mahasiswa di sistem informasi penerimaan mahasiswa</i>	<i>Field data mahasiswa di sistem informasi akademik</i>	<i>Field transformasi data mahasiswa</i>
no_pendaftaran	no_pendaftaran nim	no_pendaftaran
nisn		
nama_pendaftar	nama_mahasiswa	nama_mahasiswa
periode_pendaftaran		
alamat	alamat_mahasiswa	alamat_mahasiswa
tgl_daftar	tgl_daftar	tgl_daftar
angkatan	angkatan	angkatan
kode_prodi	kode_prodi	kode_prodi
tanggal_lahir	tanggal_lahir	tanggal_lahir
tempat_lahir	tempat_lahir	tempat_lahir
email	email	email
no_telp	no_telp	no_telp
agama	agama	agama
jenis_kelamin	jenis_kelamin	jenis_kelamin
nama_sma	nama_sma	nama_sma
tahun_tamat_sma	tahun_tamat_sma	tahun_tamat_sma
no_ijasah_sma	no_ijasah_sma	no_ijasah_sma
tanggal_ijasah	tanggal_ijasah_sma	tanggal_ijasah_sma
nilai_uan_sma	nilai_uan_sma	nilai_uan_sma
	beasiswa_mahasiswa	
	dosen_pembimbing_akd	

Hasil pemetaan data mahasiswa untuk penerapan *web services* seperti pada Tabel II, memperlihatkan sejumlah *field* data yang dapat ditransformasi dari data calon mahasiswa pada sistem informasi penerimaan mahasiswa ke data mahasiswa pada sistem informasi akademik.

Sesuai arsitektur *web services* yang ditunjukkan pada Gambar 1, maka ditentukan peran sistem informasi untuk transformasi data mahasiswa yaitu

1. Sistem informasi penerimaan mahasiswa sebagai *Server Web Services* yang menyediakan layanan akses data calon mahasiswa yang menjadi data mahasiswa yang tersimpan di sistem informasi akademik

2. Sistem informasi akademik sebagai *Client Web Services* yang mendapat akses data calon mahasiswa pada sistem informasi penerimaan mahasiswa, kemudian melengkapi data calon mahasiswa untuk menjadi data mahasiswa yang tersimpan di sistem informasi akademik

3. Layanan data calon mahasiswa yang terdapat di sistem informasi penerimaan mahasiswa dapat pula diakses melalui web oleh *Client Application*.

Web services untuk transformasi data mahasiswa menghasilkan transformasi data mahasiswa yang sesuai dari data asal yaitu sistem informasi penerimaan mahasiswa dan data hasil di sistem informasi akademik seperti ditunjukkan pada Tabel III.

TABEL III.

DATA TRANSFORMASI WEB SERVICES

<i>Field transformasi data mahasiswa</i>	<i>Data calon mahasiswa di sistem informasi penerimaan mahasiswa</i>	<i>Data mahasiswa di sistem informasi akademik</i>
no_pendaftaran	1800028	1800028
nama_mahasiswa	TRI WULANDARI	TRI WULANDARI
alamat_mahasiswa	KARANGAN BARU	KARANGAN BARU
tgl_daftar	2018-01-02	2018-01-02
angkatan	2018	2018
kode_prodi	SI-S1	SI-S1
tanggal_lahir	1997-07-04	1997-07-04
tempat_lahir	SLEMAN	SLEMAN
email	twulandari@gmail.com	twulandari@gmail.com
no_telp	89650860000	89650860000
agama	Islam	Islam
jenis_kelamin	P	P
nama_sma	SMKN 1 DEPOK	SMKN 1 DEPOK
tahun_tamat_sma	2017	2017
no_ijasah_sma	MA-06122000000	MA-06122000000
tanggal_ijasah_sma	2017-05-02	2017-05-02
nilai_uan_sma	428	428

Ketika *web services* berkerja didapatkan tampilan dari aplikasi seperti ditunjukkan pada Gambar 4 - 6.

```
{
  "_r": [
    {
      "no_pendaftaran": "1800028",
      "nama_mahasiswa": "TRI WULANDARI",
      "alamat_mahasiswa": "KARANGAN BARU",
      "tgl_daftar": "2018-01-02",
      "angkatan": "2018",
      "kode_prodi": "SI-S1",
      "tanggal_lahir": "1997-07-04",
      "tempat_lahir": "SLEMAN",
      "email": "twulandari@gmail.com",
      "no_telp": "89650860000",
      "agama": "Islam",
      "jenis_kelamin": "P",
      "nama_sma": "SMKN 1 DEPOK",
      "tahun_tamat_sma": "2017",
      "no_ijasah_sma": "MA-06122000000",
      "tanggal_ijasah_sma": "2017-05-02",
      "nilai_uan_sma": "428"
    }
  ]
}
```

Gambar 4. Ouput Data Hasil Transformasi pada *Web Services* JSON.

JSON Raw Data Headers

Copy

Response Headers

Connection Keep-Alive
 Content-Length 454
 Content-Type application/json
 Date Tue, 12 Feb 2019 04:14:33 GMT
 Keep-Alive timeout=5, max=99
 Server Apache/2.4.10 (Win32) OpenSSL/1.0.1i PHP/5.5.15
 X-Powered-By PHP/5.5.15

Request Headers

Accept text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
 Accept-Encoding gzip, deflate
 Accept-Language en-US,en;q=0.5
 Connection keep-alive
 Content-Length 494
 Content-Type application/x-www-form-urlencoded
 DNT 1
 Host localhost
 Referer http://localhost/websevice/websevice-2018/webclient/form.html
 Upgrade-Insecure-Requests 1
 User-Agent Mozilla/5.0 (Windows NT 6.1; WOW64; rv:65.0) Gecko/20100101 Firefox/65.0

Gambar 5. Request / Response Header HTML.

Daftar Calon Mahasiswa | Cari Calon Mahasiswa | Daftar Mahasiswa |

Insert Data Mahasiswa

NO PENDAFTARAN

Gambar 6. Cari Calon Mahasiswa.

Daftar Calon Mahasiswa | Cari Calon Mahasiswa | Daftar Mahasiswa |

Daftar Calon Mahasiswa

ID	NO PENDAFTARAN	NISN	NAMA PENDAFTAR	PERIODE PENDAFTARAN	ALAMAT
2	1800028		TRI WULANDARI	Gelombang 1 Reguler	KARANGAN BARU
12	1800051		LOCALHEART	Gelombang 1 Reguler	REJOWINGUN, KOTAGEDE.
13	1800053		GITA PERDANI	Gelombang 1 Reguler	BRONGKOLBENINGSRI

Gambar 7. Daftar Calon Mahasiswa.

Daftar Calon Mahasiswa | Cari Calon Mahasiswa | Daftar Mahasiswa |

Daftar Calon Mahasiswa

ID	NIM	NO PENDAFTARAN	NAMA MAHASISWA	ALAMAT MAHASISWA	TGL DAFTAR	ANGKATAN	KODE PRODI
25	185610001	1800028	TRI WULANDARI	KARANGAN BARU	2018-01-02	2018	SI-S1
23	185410005	1800050	NESYA ROGANWATI	NGLENCIS, SITIMULYO	2018-01-09	2018	TI-S1
22	185610006	1800049	PUTRI DWI RETNA ANJANI	PUCUNG PENDOWOHARJO	2018-01-09	2018	SI-S1

Gambar 8. Daftar Mahasiswa.

Daftar Calon Mahasiswa | Cari Calon Mahasiswa | Daftar Mahasiswa |

Insert Data Mahasiswa

NO PENDAFTARAN

NIM

Nama Mahasiswa

Alamat Mahasiswa

Tanggal Daftar

Tahun Angkatan

Program Studi

Tanggal Lahir

Tempat Lahir

Email

No. Telpn

Agama

Jenis Kelamin

Nama SMA

Tahun Tamat SMA

No. Ijazah SMA

Tanggal Ijazah SMA

Nilai UAN SMA

Beasiswa Mahasiswa

Dosen Pembimbing Akademik

Gambar 9. Insert Data Mahasiswa (Transformasi menjadi data mahasiswa).

IV. KESIMPULAN

Sebagai simpulan dari penelitian ini adalah bahwa pengembangan layanan web (*web services*) dengan format JSON sebagai sarana berbagi data / informasi antar sistem informasi (sistem informasi penerimaan mahasiswa dengan sistem informasi akademik) berhasil dikembangkan dan digunakan. *Web services* pada penelitian ini mentransformasi data calon mahasiswa pada sistem informasi penerimaan mahasiswa baru ke data mahasiswa pada sistem informasi akademik, dengan menghasilkan transformasi data mahasiswa yang bersesuaian antar sistem informasi tersebut. *Web services* yang dikembangkan berdasarkan arsitektur *web services* yang terdiri atas mesin yang terdapat pada jaringan yaitu *Server Web Services*, *Client Web Services*, *Client Application*

Sebagai saran untuk penelitian berikutnya adalah pengembangan layanan web (*web services*) dengan format JSON dapat dikaji

kembali sesuai dengan perkembangan teknologi yang berkembang pada penelitian berikutnya dengan metode yang relevan dengan pengembangan layanan web (*web services*), mengkaji efisiensi dan pengamanan dari pengembangan layanan web (*web services*).

UCAPAN TERIMA KASIH

Dalam penelitian tentunya tidak terlepas dari dukungan berbagai pihak yang telah membantu dan mendukung, maka pada kesempatan ini peneliti mengucapkan banyak terimakasih kepada pihak yang telah membantu penyusunan laporan ini yaitu dari pihak Pusat Penelitian dan PPM, STMIK AKAKOM Yogyakarta.

REFERENSI

- [1] A.B. Warsito, A. Ananda and D. Triyanjaya, "Penerapan Data JSON Untuk Mendukung Pengembangan Aplikasi Pada Perguruan Tinggi Dengan Teknik Restfull Dan Web Service", *Technomedia Journal (TMJ)*, vol.2, no.1, pp.26-36, Agustus 2017
- [2] A. Wijanarko and D. Ngafifuddin, "Web Service untuk Integrasi Data dalam Pengelolaan Data Potensi Prestasi Mahasiswa STMIK Amikom Purwokerto", *Teknikom*, vol. 1, no. 2, pp.51-58, 2017
- [3] D. Soares and L. Amaral, "Reflections on the Concept of Interoperability in Information Systems", In *Proceedings of the 16th International Conference on Enterprise Information Systems (ICEIS)*, 2014, vol.1, pages 331-339
- [4] D.E. Sondakh, S.R. Pungus, P. Runtukahu and R.Saroinsong, "Implementasi Teknologi Web Services Pada Aplikasi Pencarian Taksi", *Cogito Smart Journal*, vol.2, no.1, pp.1-13, Juni 2016
- [5] E. Sutanta and K. Mustofa, "Kebutuhan Web Service Untuk Sinkronisasi Data Antar Sistem Informasi Dalam E-Gov di Pemkab Bantul Yogyakarta", *Jurnal Teknologi Informasi dan Komunikasi*, vol.1, no.1, pp.31-46, Juni 2013
- [6] H. Deviana, "Penerapan XML Web service Pada Sistem Distribusi Barang", *Jurnal Generic*, vol. 6, no. 2, pp. 61-70, Juli 2011
- [7] M.A. Rahman, I. Kuswardayan, dan R.R. Hariadi, "Perancangan dan Implementasi RESTful Web Service untuk Game Sosial Food Merchant Saga pada Perangkat Android", *Jurnal Teknik Pomits*, vol. 2, no. 1, pp.1-4, 2013
- [8] R. Hidayat and A. Ashari, "Penerapan Teknologi Web Service Untuk Integrasi Layanan Puskesmas dan Rumah Sakit", *Berkala Ilmiah MIPA - Journal of Mathematics and Natural Sciences*, vol.23, no.1, pp.64-77, 2013
- [9] *Standard ECMA-404 2nd Edition December 2017 The JSON Data Interchange Syntax*, ECMA International, 2017
- [10] U.M. Qibtiyah and S. Rahayu, "Implementasi JSON Web Service pada Aplikasi Digital Library Politeknik Sukabumi", *JTERA - Jurnal Teknologi Rekayasa*, vol.2, no.1, pp.9-16, Juni 2017
- [11] Y. Herdiana, "Aplikasi Rumus Matematika SMA Berbasis Mobile", Undergraduate Theses from JBPTUNIKOMPP - Digital library - Perpustakaan Pusat Unikom, Indonesia, Maret 2014, Available: <https://elib.unikom.ac.id/gdl.php?mod=browse&op=read&id=jbptunikompp-gdl-yanaherdia-32964>