



Jurnal Ilmiah Informatika (JIF)

| ISSN (Print) 2337-8379 | ISSN (Online) 2615-1049 |

Jurnal online di akses di <http://ejournal.upbatam.ac.id/index.php/jif>



RESTful Web Service untuk Integrasi Sistem Akademik dan Perpustakaan Universitas Perjuangan

Randi Rizal^a, Alam Rahmatulloh^b

^a Teknik Informatika, Fakultas Teknik, Universitas Perjuangan, Jalan Pembela Tanah Air, Tasikmalaya

^b Informatika, Fakultas Teknik, Universitas Siliwangi, Jalan Siliwangi, Tasikmalaya

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 18 Februari 2019

Revisi Akhir: 20 Februari 2019

Diterbitkan Online: 22 Maret 2019

KATA KUNCI

Akademik, JSON, Perpustakaan, REST, *Web Service*

KORESPONDENSI

No HP: 085223519009

E-mail: alam@unsil.ac.id

A B S T R A C T

Perjuangan University is developing an information system in its campus, but the system is partially built. This has resulted in its own problems, there is no connectivity between systems and the difficulty of developing data integration. So that in the business process that occurs many experience difficulties in data synchronization and the amount of duplicate data that must be entered. One solution to overcome this problem is to utilize web service technology. The web service architecture used is REST. REST as a bridge that is able to integrate data in a stateless format. The results of the application of the web service as a backend system are able to integrate the existing academic information systems and library information systems without changing existing structures, programs or architecture. Format data exchange between systems using the JSON format. In addition, officers do not need to input and verify data several times, when the data in the library information system changes automatically in the academic information system the changes are immediately read.

1. PENDAHULUAN

Universitas Perjuangan (Unper) merupakan salah satu perguruan tinggi swasta yang berada di Kota Tasikmalaya telah mengalami banyak perkembangan di bidang Teknologi Informasi dan Komunikasi (TIK). Untuk mendukung kegiatan administrasi akademik, pengolahan data internal dan layanan informasi publik telah dikembangkan beberapa sistem informasi. Dibawah Yayasan Universitas Siliwangi, Unper mengadopsi pengembangan-pengembangan IT Universitas Siliwangi. Dimulai dari pengembangan sistem informasi laboratorium [1], sistem informasi akademik [2] [3], sistem informasi perpustakaan, sistem informasi manajemen anggaran [4], serta analisis dan pengembangan TIK [5] [6]. Namun sistem informasi tersebut belum terintegrasi satu sama lainnya sehingga timbul permasalahan diantaranya terjadi duplikasi data. Misalnya, pada sistem informasi perpustakaan seorang mahasiswa telah mengembalikan buku pinjaman, tetapi pada sistem akademik terbaca belum mengembalikan buku pinjaman sehingga proses penginputan data peminjaman dilakukan berulang.

Dengan adanya beragam aplikasi yang digunakan di setiap unit, perlu adanya integrasi data antar aplikasi. Integrasi dapat diartikan adanya saling keterkaitan antara satu sistem dengan

sistem yang lain. Pada penelitian [7] penggunaan *web service* mampu mengatasi permasalahan *interoperability* dan mengintegrasikan sistem yang berbeda. Saat ini sistem yang paling banyak digunakan oleh mahasiswa adalah sistem informasi akademik dan sistem informasi perpustakaan. Sehingga integrasi yang akan dilakukan pada penelitian ini yaitu sistem informasi akademik dengan sistem informasi perpustakaan, dengan memanfaatkan teknologi *web service*.

Web service memiliki kelebihan yaitu dapat melakukan pertukaran data lintas platform dan memiliki bahasa independent [8]. Arsitektur *web service* yang paling banyak digunakan adalah *Simple Object Access Protocol (SOAP)* dan *Representational State Transfer (REST)* [9].

Sistem akademik dengan perpustakaan mempunyai bahasa dan arsitektur yang berbeda. Arsitektur *web service* yang digunakan adalah REST, karena berdasarkan beberapa penelitian REST lebih baik dibanding SOAP [10]. RESTful *web service* dapat menggunakan pertukaran data berformat JSON maupun eXtensible Markup Language (XML).

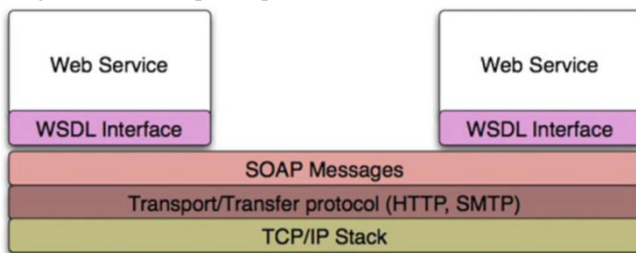
2. TINJAUAN PUSTAKA

2.1 Integrasi Data

Integrasi aplikasi *enterprise* adalah pendekatan strategis dalam menggabungkan sistem informasi bersama-sama dan dapat mendukung kemampuan untuk bertukar informasi atau data secara langsung. Integrasi aplikasi merupakan proses integrasi dari beberapa aplikasi yang bertujuan untuk berbagi data dan proses yang dapat dilakukan tanpa merubah aplikasi-aplikasi yang telah ada [11].

2.2 Web Service

Web service merupakan kumpulan suatu layanan berbasis web dengan menggunakan jaringan protokol HTTP [12], layanan tersebut dapat diakses dan dimanfaatkan oleh pengguna dengan bahasa pemrograman, arsitektur dan sistem operasi yang berbeda (*interoperability*) [13]. *Web service* harus bersifat *stateless*, sehingga dapat dibaca dan di akses multi-platform [14]. Arsitektur pada web service dapat dibangun menggunakan Simple Object Access Protocol (SOAP) atau REpresentational State Transfer (REST), sementara service tersebut direpresentasikan dalam bentuk format teks, JSON atau XML [15]. Pada umumnya formatnya menggunakan JSON dan XML. Gambar 1 menunjukkan struktur logis dari tumpukan komunikasi untuk layanan Web. Adapun perbedaan SOAP dengan REST ditampilkan pada Table 1.



Gambar. 1. Struktur logis tumpukan komunikasi layanan web

Tabel 1. Perbedaan SOAP dan REST

Uraian	SOAP	REST
Protokol komunikasi	HTTP, HTTPS, SMTP, FTP	HTTP, HTTPS
Penggunaan bandwidth	Dalam jumlah <i>request</i> yang banyak, relatif boros bandwidth. Hal ini karena banyaknya markup dalam penulisan format XML	Relatif hemat bandwidth, karena markup-markup ekstra seperti pada XML tidak dipakai
Tren penggunaan	Banyak mulai beralih ke REST, meski masih tetap ada yang mempertahankan, misalnya untuk integrasi aplikasi ke sistem legasi pada sebuah perusahaan.	Mulai populer, banyak dipakai oleh penyedia web servis terkemuka, seperti twitter, yahoo!, flickr, bloglines, technorati, google, amazon, eBay, dsb
Aturan penulisan	Ketat, mengikuti spesifikasi XML (SOAP v1.2)	Tidak ada spesifikasi khusus
Format respon	XML dengan spesifikasi SOAP. Agak sulit bagi kita untuk membaca langsung dan memahaminya.	XML, JSON, atau format plain teks lainnya. Hal ini memudahkan penerima respon membaca dan memahaminya.

Uraian	SOAP	REST
Attachment file	Bisa (karena dapat mengembalikan respon dalam format <i>binary</i>)	Tidak bisa
Sifat <i>web service</i> pada umumnya	Tertutup, lebih ditujukan untuk vendor atau perusahaan tertentu	Terbuka, bisa diakses siapa saja
<i>Caching</i> web	Relatif sulit	Mudah, karena menggunakan URI
Penggunaan standar	Standar lama (XML, HTTP) dan baru (SOAP) digunakan bersamaan	Standar yang sudah ada, seperti XML dan HTTP
Tool pengembangan	Banyak, baik komersial maupun opensource	Beberapa, karena tidak begitu dibutuhkan
Tool manajemen	Perlu, bahkan kadang harganya mahal	Menggunakan tool yang sudah ada pada sistem jaringan
Ekstensibel	Bisa, banyak ekstensi termasuk standar WS-*	Relatif tidak ekstensibel
Kemudahan implementasi	Mudah jika kita sudah memiliki lingkungan berbasis SOAP	Mudah

2.3 RESTful

Representational State Transfer yang disingkat REST yang merupakan gaya arsitektur untuk penerapan web service dalam menerapkan konsep perpindahan antar state [16]. REST populer karena kesederhanaannya, sebagai lawan menciptakan standar baru, kerangka kerja dan teknologi.

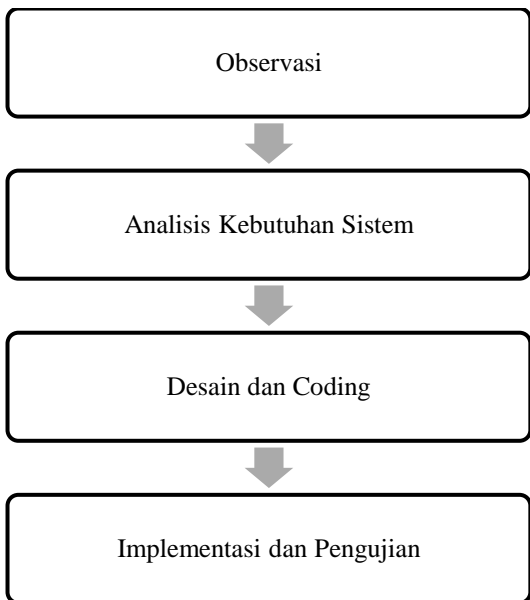
Keuntungan REST: interaksi berbasis REST menggunakan *Hypertext Transfer Protocol (HTTP)* internet yang merupakan hal yang sudah umum digunakan. Contoh dari pengaturan ini adalah interaksi berbasis REST semua berkomunikasi status mereka menggunakan kode status HTTP standar. Jadi, 404 berarti sumber daya yang diminta tidak ditemukan; kode 401 berarti permintaan tidak diotorisasi; kode 200 berarti semuanya OK; dan 500 berarti ada kesalahan aplikasi yang tidak dapat dipulihkan pada server.

Demikian pula, detail seperti enkripsi dan integritas transport data diselesaikan bukan dengan menambahkan kerangka kerja atau teknologi baru, tetapi dengan mengandalkan enkripsi Secure Sockets Layer (SSL) dan Transport Layer Security (TLS) yang terkenal. Jadi, seluruh arsitektur REST dibangun di atas konsep yang sudah dikenal sebagian besar pengembang.

Namun, manfaat REST menggunakan konstruk HTTP juga menciptakan batasan. Banyak keterbatasan HTTP juga berubah menjadi kekurangan gaya arsitektur REST. Misalnya, HTTP tidak menyimpan informasi berbasis negara di antara siklus permintaan-respons, yang berarti aplikasi berbasis REST harus tanpa kewarganegaraan dan tugas manajemen negara apa pun harus dilakukan oleh klien [17].

3. METODOLOGI

Tahapan penelitian terdiri dari empat tahapan dimulai dari observasi sistem akademik dan perpustakaan yang ada, kemudian analisa kebutuhan untuk integrasi antar sistem, desain dan *coding*, terakhir implementasi serta pengujian sistem seperti yang dapat dilihat pada Gambar 2.



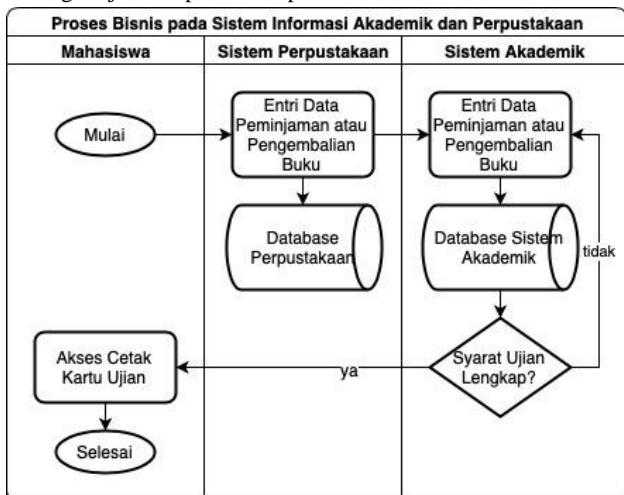
Gambar. 2. Metode Penelitian

3.1 Observasi

Observasi dilakukan terhadap sistem informasi akademik yang menggunakan Bahasa pemrograman PHP dengan framework Laravel. Databasenya menggunakan MySQL dengan web server Apache. Sementara pada sistem perpustakaan menggunakan framework SLiMS (Senayan Library Management System) yang mempunyai arsitektur dan source code yang berbeda.

3.2 Analisis Kebutuhan Sistem

Berdasarkan hasil observasi yang telah dilakukan, pada sistem informasi akademik terdapat Nomor Induk Mahasiswa (NIM) sebagai data unik, begitu juga pada sistem perpustakaan NIM dijadikan sebagai kunci utama. Adapun proses bisnis yang sedang berjalan dapat dilihat pada Gambar 3.



Gambar. 3. Bisnis Proses Pengecekan Syarat Ujian

Gambar 2 menunjukkan pada proses bisnis pengecekan kartu ujian sebagai syarat ujian salah satunya peminjaman buku perpustakaan harus dikembalikan. Proses entri data buku dilakukan dua kali yaitu pada sistem perpustakaan begitu juga

pada sistm akademik. Hal tersebut menjadi kendala serta mengakibatkan duplikasi data.

Setelah dilakukan analisis sistem, tahapan berikutnya dilakukan analisis kebutuhan perangkat lunak. Perangkat lunak yang diperlukan dalam pengembangan service yang dibutuhkan untuk integrasi sistem ditampilkan pada Tabel 2.

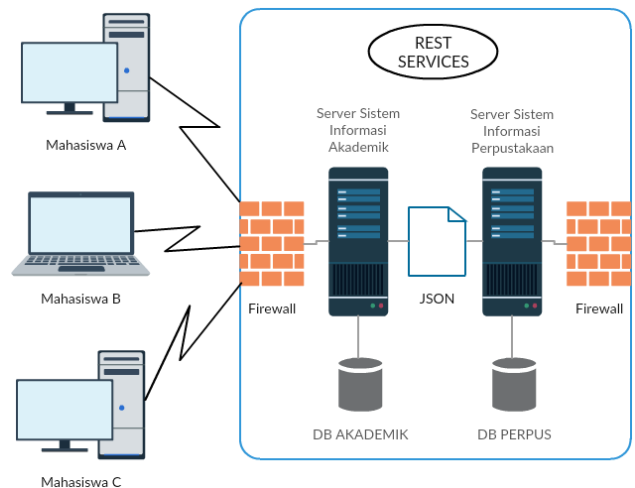
Tabel 2. Kebutuhan Perangkat Lunak

No	Software	Spesification
1.	Web Server	Apache v2.3
2.	Database Server	MariaDB v10.1.0
3.	Web Programming	PHP v7
4.	Web Browser	Safari v12.0.3

3.3 Desain dan Coding

3.3.1 Arsitektur Sistem

Perancangan arsitektur sistem web service untuk menggabungkan dua sistem yaitu sistem informasi akademik dengan sistem informasi perpustakaan dapat dilihat pada Gambar 3.



Gambar. 4. Arsitektur REST Services

Gambar 4 merupakan arsitektur usulan dalam penelitian ini, yang terdiri dari server sistem informasi akademik dan server sistem informasi perpustakaan serta tidak ada perubahan pada masing-masing aplikasi. Hanya ada penambahan web service yang menjembatani antar server dengan format pertukaran data menggunakan JSON.

3.3.2 Class Application Program Interface (API)

Application program interface dibuat dengan metode RESTful dengan format JSON untuk pertukaran data antara sistem informasi perpustakaan dengan sistem informasi akademik. Potongan kode sumber Class API dan REST Services dapat dilihat pada Gambar 5 dan proses konversi data kedalam format JSON serta pemanggilan class api dapat dilihat pada Gambar 6.

```

8 class api
9 {
10     private $db;
11     function __construct()
12     {
13         $conf = json_decode(file_get_contents('configuration.json'), TRUE);
14         $this->db = new mysqli($conf["host"], $conf["user"], $conf["password"],
15             $conf["database"]);
16     }
17     function __destruct()
18     {
19         $this->db->close();
20     }
21     function get($params)
22     {
23         $query = 'SELECT u.member_id AS id'
24             . ', u.loan_date AS loan_date'
25             . ', u.due_date AS due_date'
26             . ', u.item_code AS item_code'
27             . ', u.title AS judul'
28             . ', u.is_return AS is_return'
29             . 'FROM loan AS u'
30             . 'LEFT OUTER JOIN item i on i.item_code = u.item_code'
31             . 'LEFT OUTER JOIN biblio b on b.biblio_id = u.biblio_id'
32             . '($params[id] = '?' : ' WHERE is_return = 0 and u.member_id = \'' .
33             $this->db->real_escape_string($params['id']) . '\')'
34             . ' ORDER BY u.due_date';
35         $list = array();
36         $result = $this->db->query($query);
37         while ($row = $result->fetch_assoc())
38         {
39             $list[] = $row;
40         }
41         return $list;
42     }
43 }

```

Gambar. 5. Application Program Interface (API)

```

21 require_once 'api.php';
22
23 // creates a new instance of the api class
24 $api = new api();
25
26 // message to return
27 $message = array();
28
29 switch($_POST["action"])
30 {
31     case 'get':
32         $params = array();
33         $params['id'] = isset($_POST["id"]) ? $_POST["id"] : '';
34         if (is_array($data = $api->get($params))) {
35             $message["code"] = "0";
36             $message["data"] = $data;
37         } else {
38             $message["code"] = "1";
39             $message["message"] = "Error on get method";
40         }
41         break;
42
43     default:
44         $message["code"] = "1";
45         $message["message"] = "Unknown method " . $_POST["action"];
46         break;
47 }
48
49 //the JSON message
50 header('Content-type: application/json; charset=utf-8');
51 echo json_encode($message, JSON_PRETTY_PRINT | JSON_UNESCAPED_UNICODE
52 | JSON_UNESCAPED_SLASHED);

```

Gambar. 6. REST Services

```

        "item_code": "00171",
        "judul": "OBAT ASLI INDONESIA",
        "is_return": "0"
    },
    {
        "id": "08046",
        "loan_date": "2010-04-16",
        "due_date": "2010-04-22",
        "item code": "00012",
        "judul": "Obat-Obat Penting;
        Khasiat, Penggunaan dan Efek-Efek Sampingnya",
        "is_return": "0"
    }
]

```

Gambar. 7. REST Services

4.2. Pengujian REST Services pada Sistem

Pengujian dilakukan dengan beberapa skenario yaitu:

1. Jika server sistem perpustakaan tidak tersedia (*down*), atau konektivitas antar server terganggu/tidak terjangkau (*network unreachable*)
2. Jika mahasiswa belum mengembalikan barang perpustakaan (buku)
3. Jika mahasiswa sudah mengembalikan buku/tidak ada buku perpustakaan yang dipinjam

Skenario ke-1: server perpustakaan dimatikan.

Cetak Kartu UTS

No.	Mata Kuliah	Kehadiran
1	Farmakologi Dasar dan Toksikologi	100.00 %
2	Biokimia	100.00 %
	Praktikum	100.00 %
3	Biologi Sel dan Molekuler	100.00 %
4	Kimia Analisis	100.00 %
	Praktikum	100.00 %
5	Teknologi Form. Sed. Solid dan Liquid	100.00 %
	Praktikum	100.00 %
6	Farmakognosi Farmasi	100.00 %
	Praktikum	100.00 %
7	Metabolisme	100.00 %

Tidak dapat terhubung ke server perpustakaan
Syarat cetak kartu UTS belum terpenuhi

Cek Ulang Perpustakaan
Batal
Cetak

Gambar. 8. Respon Server Perpustakaan Down

4. HASIL DAN PEMBAHASAN

4.1. Implementasi

Contoh respon REST Services yang dihasilkan dalam bentuk format JSON yang dapat dilihat pada Gambar 7. Web service memberikan respon dalam bentuk JSON dengan data buku yang belum dikembalikan, lengkap dengan tanggal peminjaman, judul, kode sampai tanggal pengembalian.

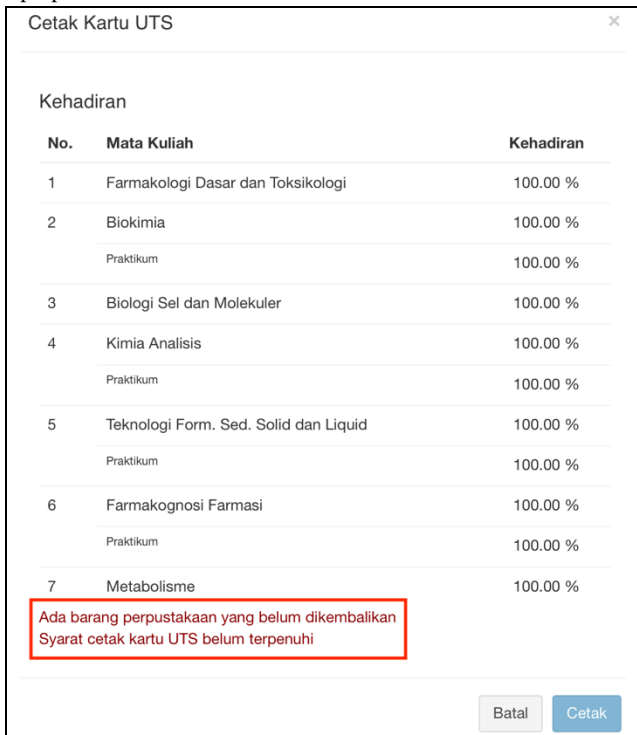
```

{
  "code": "0",
  "data": [
    {
      "id": "08046",
      "loan_date": "2010-03-18",
      "due_date": "2010-03-24",
      "item_code": "6107alip",
      "judul": "Pengantar Konsep Dasar
      Keperawatan",
      "is_return": "0"
    },
    {
      "id": "08046",
      "loan_date": "2010-03-19",
      "due_date": "2010-03-25",

```

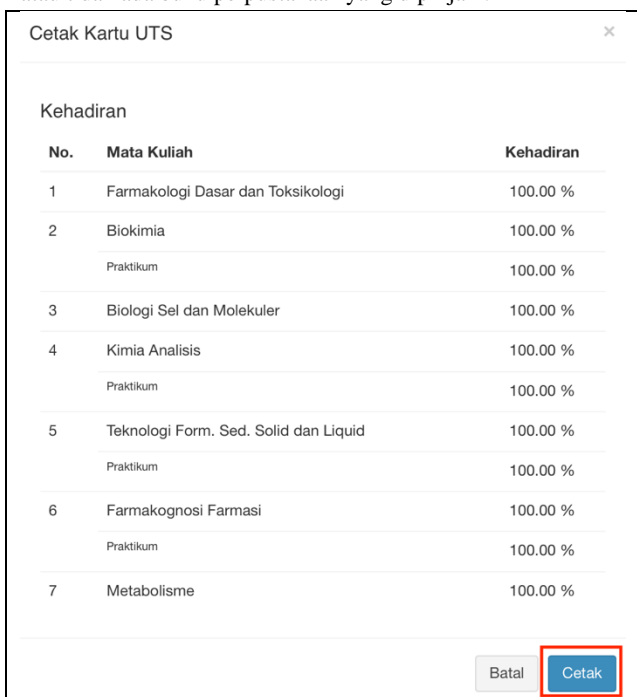
Gambar 8 menunjukkan respon dalam bentuk JSON sudah diolah dan dimanipulasi aplikasi kedalam bentuk informasi yang muncul dalam laman pencetakan kartu ujian UTS maupun UAS. Skenario ke-1 dilakukan untuk menguji apabila server perpustakaan tidak dapat diakses atau *down*, *web service* tetap dapat meng-*handle* permasalahan tersebut serta muncul *warning* untuk melakukan pengecekan pada *server* perpustakaan.

Skenario ke-2: mahasiswa belum mengembalikan buku perpustakaan.



Gambar. 9. Respon Buku Perpustakaan Belum Dikembalikan

Skenario ke-3: mahasiswa telah mengembalikan semua buku atau tidak ada buku perpustakaan yang dipinjam.



Gambar. 10. Respon Buku Perpustakaan Sudah Dikembalikan

Pada skenario ke-2 Gambar 9 memberikan informasi yang berbeda dengan Gambar 8 yaitu kondisi server perpustakaan dapat diakses dan mahasiswa tersebut mempunyai data buku yang belum dikembalikan. Sehingga walaupun data kehadiran 100% ada, namun jika data buku belum dikembalikan mahasiswa tetap tidak dapat melakukan pencetakan kartu ujian.

Sementara skenario ke-3 pada Gambar 10 menunjukkan kondisi mahasiswa tersebut sudah mengembalikan buku perpustakaan sehingga tidak muncul informasi (*warning*) sehingga tombol pencetakan pun aktif dan dapat dicetak.

5. KESIMPULAN

Penerapan teknologi *web service* dengan menggunakan arsitektur REST pada sistem informasi akademik dan sistem informasi perpustakaan mampu mengintegrasikan kedua sistem tersebut. Sehingga proses input dan verifikasi data hanya dilakukan satu kali, hal tersebut mengatasi terjadinya duplikasi data dan mengurangi pekerjaan input data. Format pertukaran data antar sistem menggunakan format JSON yang tidak bernegara (*stateless*), sehingga memudahkan untuk dapat diakses oleh bahasa pemrograman apapun, arsitektur maupun sistem operasi yang berbeda.

Pada penelitian ini belum ada diskusi lebih mendalam mengenai keamanan data dan otentikasi antar sistem, sehingga perlu ada penelitian kembali mengenai keamanan dan otentikasi pada *web service* dengan menerapkan konsep *single sign on (SSO)*.

DAFTAR PUSTAKA

- [1] E. W. Hidayat, A. Rahmatulloh dan D. R. Setiawan, "Pengembangan Sistem Informasi Laboratorium Teknik Informatika (SILABTI) Dengan Arsitektur Model View Controller," *SITROTIKA*, vol. 6, no. 2, 2010.
- [2] E. W. Hidayat dan A. Rahmatulloh, "Optimalisasi Kinerja Sistem Informasi Akademik Universitas Siliwangi menggunakan Memcached dan Mirror Server," *Konferensi Nasional Sistem Informasi*, 2014.
- [3] E. W. Hidayat dan A. Rahmatulloh, "Optimasi Server SIMAK Menggunakan Memcached dan Mirror Server Untuk Meningkatkan Kecepatan Akses Layanan Akademik Universitas Siliwangi," *S@ CIES*, vol. 5, no. 2, 2015.
- [4] A. Rahmatulloh dan Husen, "Sistem Informasi Manajemen Anggaran (Simangga) Perguruan Tinggi Berbasis Web (Studi Kasus: Universitas Siliwangi)," *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 3, no. 2, pp. 89-95, 2017.
- [5] A. Rahmatulloh dan F. MSN, "Implementasi Load Balancing Web Server menggunakan Haproxy dan Sinkronisasi File pada Sistem Informasi Akademik Universitas Siliwangi," *Jurnal Teknologi dan Sistem Informasi*, vol. 3, no. 2, 2017.
- [6] E. W. Hidayat, F. M. SN dan A. Rahmatulloh, "ANALISIS SERVICE QUALITY UNTUK MENGUKUR TINGKAT KEPUASAN PENGGUNA LAYANAN TIK DI UNIVERSITAS SILIWANGI," *SEMNASTEKNOMEDIA ONLINE*, vol. 5, no. 1, 2017.
- [7] R. Gunawan dan A. Rahmatulloh, "Implementasi Web Service pada Sistem Host-To-Host Pembayaran Biaya Akademik," *Setrum: Sistem Kendali-Tenaga-Elektronika-Telekomunikasi-Komputer*, pp. 320-328, 2019.
- [8] I. M. Siregar dan J. Purba, *Membongkar Teknologi*

Pemrograman Web Service, Bandung: AplyIT, 2012.

- [9] Y. Fauziah, “Aplikasi Iklan Baris Online menggunakan Arsitektur REST Web Service,” *TELEMATIKA*, Vol. %1 dari %2Vol. 9, No. 2, p. 75 – 80, JANUARI 2013.
- [10] A. Memeti, F. Imeri dan B. Cico, “REST vs. SOAP: Choosing the best web service while developing in-house web applications,” *Journal of Natural Sciences and Mathematics of UT*, vol. 2, no. 3, pp. 63-68, 2017.
- [11] J. Puustjärvi dan L. Puustjärvi, “Application Integration and Semantic Integration in Electronic Prescription Systems,” *International Journal of Computer Science Issues*, vol. 7, no. 1, pp. 1-8, 2010.
- [12] R. Fielding dan J. Reschke, “Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content,” *Internet Engineering Task Force (IETF)*, June 2014.
- [13] R. T. Fielding dan R. N. Taylor, “Principled design of the modern Web architecture,” *ACM Transactions on Internet Technology (TOIT)*, vol. 2, no. 2, pp. 115-150, 2002.
- [14] H.-y. Paik, A. L. L. M. C. Barukh dan B. B. A. Natarajan, *Web Service Implementation and Composition Techniques*, Springer, 2017.
- [15] K. Gottschalk, S. Graham, H. Kreger dan J. Snell, “Introduction to Web services architecture,” *IBM SYSTEMS JOURNAL*, vol. 41, no. No. 2, 2002.
- [16] R. T. Fielding, *Architectural Styles and the Design of Network-based Software Architectures*, University of California, Irvine, 2000.
- [17] C.-J. Su dan C.-Y. Chiang, “Enabling successful Collaboration 2.0: A REST-based Web Service and Web 2.0 technology oriented information platform for collaborative product development,” *Computers in Industry*, vol. 63, 2012.

BIODATA PENULIS



Randi Rizal, S.T., M.Kom.

Dosen Teknik Informatika, Fakultas Teknik, Universitas Perjuangan, Tasikmalaya, praktisi aktif dalam bidang IoT, security dan digital forensic. Selain sebagai pengajar juga sebagai CEO Sekolah Programmer yuCoding yang merupakan sekolah pemrograman pertama di Tasikmalaya.



Alam Rahmatulloh, S.T., M.T.

Saat ini menjadi dosen Informatika Fakultas Teknik Universitas Siliwangi Tasikmalaya, serta praktisi dan aktif dalam bidang informatika, web programming, *software security*, multimedia dan IoT. Selain pengajar, berperan aktif bidang IT di Universitas Siliwangi Tasikmalaya.