

Perancangan Arsitektur RESTful pada Integrasi Data Referensi Buku di Perpustakaan Daerah dan Perpustakaan Universitas (Studi Kasus: Kota Salatiga)

Erwien Christiano¹, Eko Sedyono²
erwinchristiant@gmail.com¹, ekosed1@yahoo.com²
Faculty of Information Technology
Satya Wacana Christian University
Jl. Diponegoro 52-60, Salatiga 50771, Indonesia

Abstract:

The library in this modern era in the advancement of information technology needs to be developed towards data integration. Network communication between libraries can be a means of good learning, effective and efficient. The problem often encountered in collaborating on the business process for integration of library data is the difference of application to the user of each part. Technology that can integrate data even though different applications / platforms on end users with web service technology. With the web service for RESTful method can bridge the integration of data exchange for book reference in library so that can be know where and where book have references to references book or document needed. Restful technology is created with java class and database as prototype and simulation. Clear and viable business processes can be integrated into an integrated information system.

Keywords: *Web service, Restful, Integration, Reference Book*

1. Pendahuluan

1.1. Latar Belakang

Teknologi internet telah maju dalam perkembangan sistem *distributed computing* dengan munculnya konsep mengenai *web services*. *Web services* merupakan salah satu paradigma baru dalam mengimplementasikan sistem terdistribusi melalui web yang menggunakan basis teknologi XML, dengan standard protokol HTTP dan SOAP. Dari standard yang digunakan dalam mendukung teknologi *web services*, maka implementasi *web services* menjanjikan banyak kemudahan dan perbaikan dalam mendukung integrasi berbagai platform system dan aplikasi, baik melalui infrastruktur intranet atau skstranet. Keuntungan utama dari integrasi sistem informasi adalah memperbaiki arus informasi dalam sebuah organisasi.

Sebuah organisasi khususnya perpustakaan sebagai salah satu institusi pelayan masyarakat akan melayani transaksi penjaminan buku dalam kesehariannya. Pemberian layanan dan tindakan dalam banyak hal akan mempengaruhi pemnjam. Semakin cepat akan semakin baik karena menyangkut peminjam dalam menemukan tempat buku referensi berada. Semakin besar jasa layanan suatu perpustakaan, akan semakin kompleks pula jenis tindakan dan layanan yang harus diberikan yang kesemuanya harus tetap dalam satu koordinasi terpadu. Karena selain memberikan layanan, perpustakaan juga harus mengelola buku baik katalog dan referensi buku untuk menentukan kaitan buku referensi itu berada di perpustakaan mana. Melihat situasi tersebut, sudah sangatlah tepat jika perpustakaan harus menggunakan

sisi kemajuan komputer, baik piranti lunak maupun perangkat kerasnya dalam upaya membantu penanganan manajemen yang sebelumnya belum dilakukan. RESTful adalah suatu model arsitektural untuk membangun solusi *enterprise* berdasarkan *service*. Secara lebih spesifik, REST berhubungan dengan pembangunan independen dari layanan bisnis yang dapat dikombinasikan menjadi proses bisnis pada level tinggi dan solusi dalam konteks *enterprise*.

Berdasarkan latar belakang yang ada, penelitian ini bertujuan untuk merancang pencarian buku referensi dengan pemanfaatan *web service* dengan metode RestFul yang dapat mengkomunikasikan data antar perpustakaan daerah dan perpustakaan universitas atau perpustakaan pada fakultas. Perancangan ini dibangun untuk memudahkan bagian pustakawan dalam melakukan pencarian lokasi buku/jurnal apa saja yang menjadi keterkaitan dalam buku referensi yang dicari/dibutuhkan.

1.2. Tinjauan Pustaka

Ada beberapa penelitian sebelum yang diambil sebagai acuan perbandingan atau membedakan dengan temuan penelitian yang dilakukan:

Penelitian lain juga dilakukan dalam perancangan *library automation* dengan metode rapid application development (RAD) menerapkan RESTful *Web service* menggunakan *AsyncTask*. Untuk hasilnya *library automation* berbasis mobile yang interaktif dan mudah digunakan oleh civitas akademik Universitas Negeri Semarang yang dapat menghasilkan menu pelacakan koleksi, riwayat koleksi yang pernah dilakukan, dan perhitungan denda keterlambatan pengembalian pustaka. [1]

Data integrasi kebutuhan harga pokok dalam suatu proses bisnis perusahaan garment menggunakan *Service Oriented Arcitecture* (SOA). Menghasilkan point –point yang dibutuhkan tiap bagian dan setiap bagian akan menginformasikan hasil dan di olah oleh *web service* untuk menghasilkan harga pokok penjualan. [2]

Untuk memperoleh Informasi penelitian perguruan tinggi harus mengunjungi web site masing-masing perguruan tinggi. Karena masing-masing perguruan tinggi mempublikasikan informasinya sendiri-sendiri. Usaha yang sudah ada untuk mengintegrasikan informasi penelitian dengan cara menyamakan format data, tetapi kurang berhasil. Informasi penelitian dapat juga dicari melalui search engine, tetapi informasi yang dihasilkan tercampur dengan informasi lain. Masalah yang ada dapat dilakukan dengan cara mengintegrasikan aplikasi dalam masing – masing bidang di perpustakaan sebagai sumber informasi penelitian dalam menerapkan *web service*. Mengintegrasikan tiga jenis basis data perpustakaan yang berbeda platform menggunakan tiga *web service* yang berbeda ke dalam satu aplikasi berbasis web.[3].

1.2.1. Buku Referensi

Kata referensi berasal dari inggris *reference* dan merupakan kata kerja to refer yang artinya menunjukkan kepada. Buku referensi adalah buku yang dapat

memberikan keterangan topik perkataan, tempat, peristiwa, data statistika, pedoman, alamat, nama orang, riwayat orang-orang terkenal. Pelayanan referensi adalah pelayanan dalam menggunakan buku-buku referensi. di perpustakaan biasanya buku-buku referensi dikumpulkan tersendiri dan di sebut “koleksi referensi” sedangkan ruang tempat penyimpanan disebut ruang referensi. Buku-buku referensi yang karena sifatnya sebagai buku penunjuk, harus selalu tersedia di perpustakaan sehingga dapat dipakai oleh setiap orang pada setiap saat.

Disebut koleksi rujukan karena merupakan sebuah buku atau kumpulan buku yang didesain untuk dikonsultasikan atau diacu dari masa ke masa untuk mencari informasi khusus tentang suatu topik, tema, peristiwa, orang, tempat ataupun kata-kata. Koleksi referensi bertujuan untuk memberikan informasi kepada pengguna, baik informasi bersifat ilmiah untuk kepentingan studi atau riset maupun informasi lain yang bersifat non-ilmiah.

Menurut Zainuddin (1996:21) dijelaskan bahwa:

A collection of books and other materials in a library useful for supplying information, kept together for convenience and generally not allowed to be circulated. (Suatu koleksi yang terdiri atas buku-buku dan bahan-bahan pustaka lain yang terdapat dalam sebuah perpustakaan yang sepenuhnya digunakan bagi penyediaan informasi, diolah untuk memudahkan penelusuran bagi pengguna dan biasanya tidak untuk dipinjamkan).

Koleksi referensi tidak digunakan untuk dibaca secara keseluruhan atau terus-menerus seperti halnya buku teks, tetapi hanya dibaca pada bagian informasi yang dibutuhkan saja. Koleksi referensi juga tidak dapat dipinjamkan untuk dibawa pulang melainkan hanya dapat dibaca saja [4].

Menurut Sulistyio Basuki, ciri-ciri koleksi referensi adalah sebagai berikut:

- a. Buku referensi ditujukan untuk keperluan konsultasi. Lazimnya hanya bagian tertentu saja yang digunakan untuk suatu kepentingan.
- b. Buku referensi tidak dimaksudkan untuk dibaca seperti buku biasa.
- c. Buku referensi serig kali terdiri dari entri yang terpotong-potong. Masing-masing entri tidak sama panjangnya. Dengan kata lain buku Universitas Sumatera Utara 10 referensi biasanya ditandai dengan pemaparan buku referens yang tidak berkesinambungan.
- d. Di perpustakaan buku referensi biasanya tidak dipinjamkan karena buku tersebut diperlukan setiap waktu untuk dikonsultasi. Pustakawan tidak dapat menduga bilamana sebuah buku referensi diperlukan sehingga penggunaannya terbatas pada ruang referensi saja.
- e. Informasi disusun untuk memudahkan penelusuran secara cepat dan menyeluruh. Susunan ini dapat menurut abjad, judul, subjek, atau kronologis disertai indeks untuk keperluan temu balik informasi.

Walaupun terdapat ciri-ciri buku referensi seperti di atas dalam praktik sehari-hari, batas antara buku referensi dengan buku biasa sangatlah berbeda. Keputusan untuk menentukan apakah sebuah buku termasuk buku referensi atau buku yang akan dipinjamkan itu tergantung pada masing-masing perpustakaan.[5]

1.2.3. Web service

Web Service adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interaction and interoperability antar sistem pada suatu jaringan . *Web service* digunakan sebagai suatu fasilitas yang menyediakan layanan (dalam bentuk informasi atau data) kepada sistem lain Hal ini dikarenakan *web services* dibangun diatas protokol-protokol yang sudah terkenal dan memiliki platform yang independent, seperti HTTP, XML, UDDI, dan WSDL [6]. SOA menggunakan protokol-protokol tersebut sebagai komponen kunci karena protokol-protokol menyediakan layanan yang dapat ditemukan dan digunakan secara dinamis, sehingga dapat berinteraksi dengan sistem tersebut melalui layanan-layanan yang disediakan. *Web service* 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]



Gambar 1. Proses *Web service* [7]

Berikut penjelasan tentang proses yang terjadi pada *web service*:

- Proses pencarian layanan (*find*). Proses ini dilakukan oleh service consumer terhadap server yang memiliki daftar layanan yang dapat digunakan serta cara untuk menggunakannya.
- Proses menggunakan layanan (*bind and invoke*). Proses ini dilakukan apabila agreement antara *service consumer* dan *service provider* telah didaftarkan pada *registry server*.
- Proses register. Proses ini dilakukan oleh *service provider* untuk mendaftarkan service yang dapat digunakan oleh service consumer pada registry server.
- Service contract*. Bagian ini adalah antarmuka service yang mendefinisikan komponen-komponen yang terdapat pada sebuah service. Service contract direpresentasikan dalam bentuk WSDL. WSDL adalah sebuah deskripsi bahasa yang dapat dimengerti oleh manusia dan mesin komputer. WSDL ditulis dengan menggunakan sintaks XML dan digunakan sebagai antarmuka untuk melakukan message exchange dalam suatu layanan. Contoh komponen-komponen yang terdapat pada *service contract* adalah nama layanan, nomor versi, header[7].

1.2.4. Bisnis Proses

Suatu perusahaan dalam perkembangan dan persaingannya harus menjadi semakin siap dan matang dalam menjalankan aktivitas bisnisnya. Untuk mencapai kondisi tersebut, sebuah perusahaan harus senantiasa mempertimbangkan dan

mengoptimasi cara mereka menjalankan bisnis dan mengubah sistem informasi dan aplikasi untuk mendukung perubahan yang dilakukan pada proses bisnis mereka. Berikut adalah beberapa definisi proses bisnis.

1. *Business process represent the unique ways in whit organization coordinate and knowledge to produce a product or service [8]*

2. *A business process is a set of logically related tasks performed to achieve a defined business outcome. Within the business process, people, wquipment, material resources, and business procedures and combined to produce a specified result [9]*

3. *Increases in consumer requirements for both product and service efficiency and effectiveness has resulted in business process reengineering (BPR). The reengineering of business processes is concerned with fundamentally rethinking and redesigning business processes to obtain dramatic and sustaining improvements in quality, cost, service, lead-times, outcomes, flexibility and innovation. [10].*

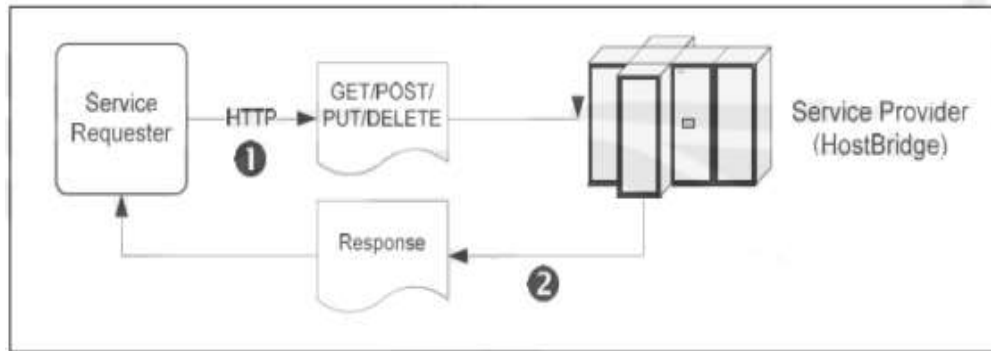
4. *Business Process* didefinisikan sebagai sebuah kumpulan dari pekerjaan-pekerjaan yang saling berhubungan untuk menyelesaikan suatu permasalahan. Sebuah *business process* dapat dipecah menjadi sekumpulan sub-proses yang mempunyai atribut masing-masing, namun tetap berkontribusi untuk melaksanakan tujuan dari proses utama. *Business process* merupakan inti dari seluruh aktivitas pada suatu perusahaan atau organisasi. Untuk mencapai tujuan perusahaan, *business process* yang akan memberdayakan seluruh sumber daya yang ada pada perusahaan. Tapi yang perlu diketahui adalah bahwa setiap bisnis memiliki proses masing-masing yang unik, sesuai dengan karakteristik dari perusahaan dan bidang usahanya, seperti proses pembuatan produk ataupun layanan baru, pengadaan *supply*, menjawab pertanyaan pelanggan, ataupun rekrutasi karyawan baru, yang tentunya memiliki perbedaan karekteristik tersendiri untuk setiap perusahaan [11].

Sistem informasi dapat membantu suatu organisasi dalam mencapai efisiensi dengan cara mengotomatisasi bagian-bagian dari proses bisnis atau dengan membantu organisasi memikirkan kembali dan mempersingkat proses bisnis tersebut, jadi dalam hal ini proses bisnis dapat diimplementasikan menjadi suatu sistem informasi berbasis komputer. Untuk keperluan tersebut maka dibutuhkan sistem analis yang bertugas mempelajari permasalahan dan kebutuhan dari organisasi untuk menentukan bagaimana orang, data, proses, komunikasi, dan teknologi informasi yang diterapkan dalam bisnis tersebut [12].

1.2.5. RestFul

Representational State Transfer yang disingkat REST yang merupakan arsitektur untuk penerapan *web service* dalam menerapkan konsep perpindahan antar *state*[13]. State disini dapat digambarkan seperti peramban meminta suatu halaman situs, di sisi server akan mengirimkan state halaman situs yang sekarang ke peramban. Navigasi melalui URL yang disediakan sama halnya dengan mengganti state dari halaman situs. Sama seperti REST bekerja, dengan bernavigasi melalui link HTTP untuk melakukan aktivitas tertentu. Seakan-akan

terjadi perpindahan state antara satu dengan yang lain. Perintah HTTP yang bisa digunakan dalam REST adalah fungsi GET, POST, PUT atau DELETE. REST WS membangun integrasi dengan cara yang lebih ringan dan sederhana, dan berfokus pada sumberdaya[14]. Dalam pengaplikasiannya, REST lebih banyak digunakan pada *web service* yang berorientasi data sumber daya. Sebutan untuk *web service* yang menerapkan arsitektur REST adalah RESTful *web service*. [15]

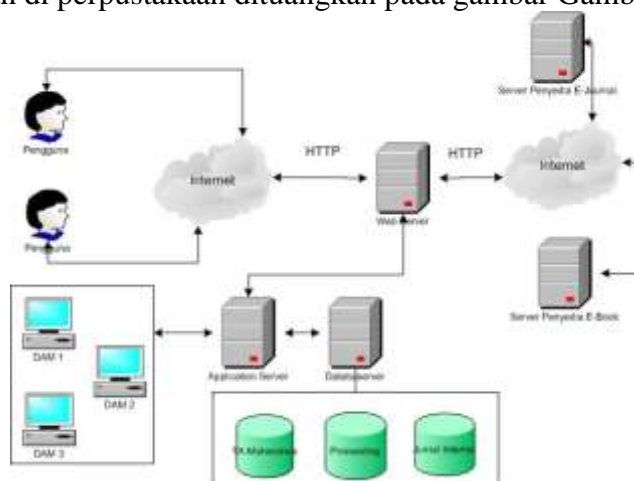


Gambar 2 : Model dasar RESTful *Web services* [16]

2. Metode Penelitian

2.1. Analisa Lingkungan Masalah

Penelitian di Perpustakaan ini karena terdapat beberapa unit dilingkungan perpustakaan yang masing - masing unit memiliki sistem yang berbeda kebutuhan tiap-tiap unit. terkumpul dari hasil wawancara mendesain arsitektur yang dibangun. Dengan tujuan untuk melihat lebih jauh berbagai komponen yang dipakai pada sistem yang digunakan adalah informasi, hardware, software manusia yang tersedia di lingkungan perpustakaan. Sistem yang menghasilkan desain fungsional. Desain sistem berupa desain konseptual dari sistem dengan tujuan yaitu menghasilkan spesifikasi yang sesuai dengan kebutuhan pengguna. Konsep arsitektur yang digunakan pada sistem pencarian di perpustakaan dituangkan pada gambar Gambar 3.

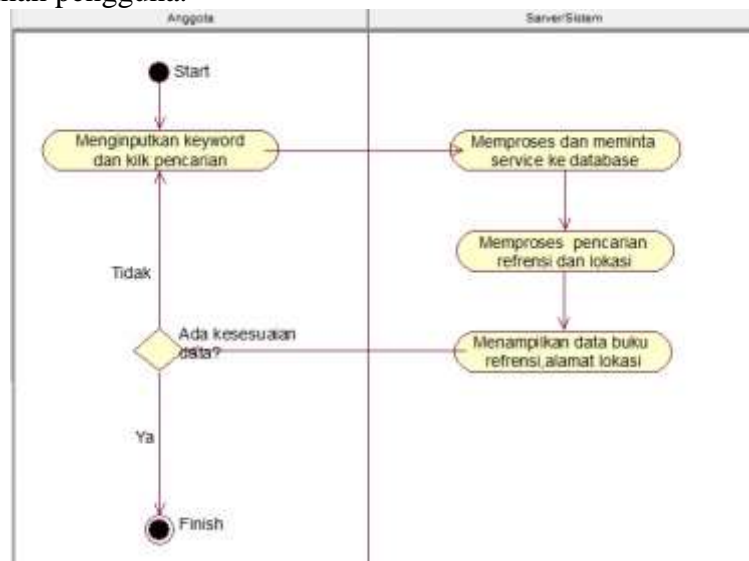


Gambar 3. Rancangan Arsitektur Perpustakaan Digital Perguruan Tinggi (Pendit, 2007)

Arsitektur tersebut terdiri atas lapisan *Business Logic* dan data *Access* berfungsi untuk memproses setiap komunikasi (*POST, GET, PUT dan DELETE*). Pada penerapannya, menerjemahkan setiap *request HTTP* dari *client*. Sedangkan mengimplementasikan fungsionalitas inti dari sistem, dan merangkum logika bisnis yang relevan. Lapisan data access berperan untuk mengekspos data berdasarkan batasan batasan yang dimiliki oleh sistem. Data atau informasi yang diekspos menggunakan format data JSON. Jason adalah bahasa independen dan berdasarkan koleksi pasangan *key/value* dan mempunyai list value. Struktur ini memungkinkan untuk digunakan dalam setiap bahasa pemrograman modern yang membuatnya menjadi pilihan yang baik untuk berkomunikasi di dunia web.[15]

2.2. Proses Bisnis Pencarian

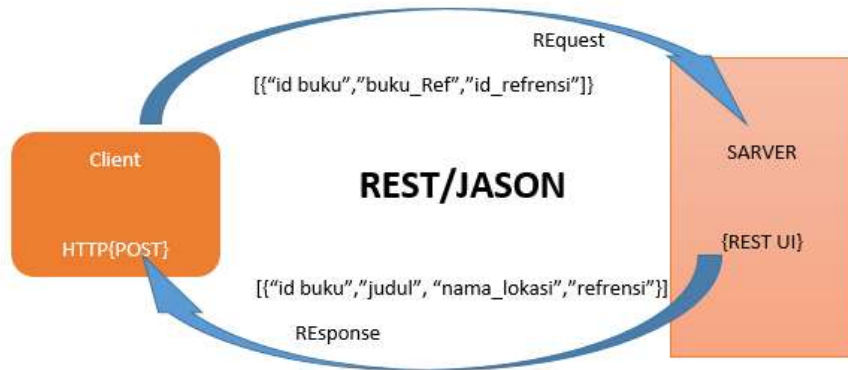
Proses bisnis utama (*primary processes*) pada perpustakaan umumnya meliputi proses-proses inti yang berupa pelayanan dasar pada pengguna. Proses bisnis ini dimulai dari pengadaan situs dan penginputan data koleksi dari berbagai sumber, pengelolaannya, sampai pada aktivitas pelayanan pengguna.



Gambar 4. Proses Bisnis Pencarian Buku Referensi

2.3. Desain Arsitektur

Desain dari *client* dan *server* yang berkomunikasi yang menggunakan protokol HTTP untuk pertukaran data. Dengan rest server yang sebagai penyedia jalur akses *resource* atau data dan dari sisi REST client yang dapat melakukan *resource* dan kemudian menampilkan atau menggunakannya baik dari XML/Jason.

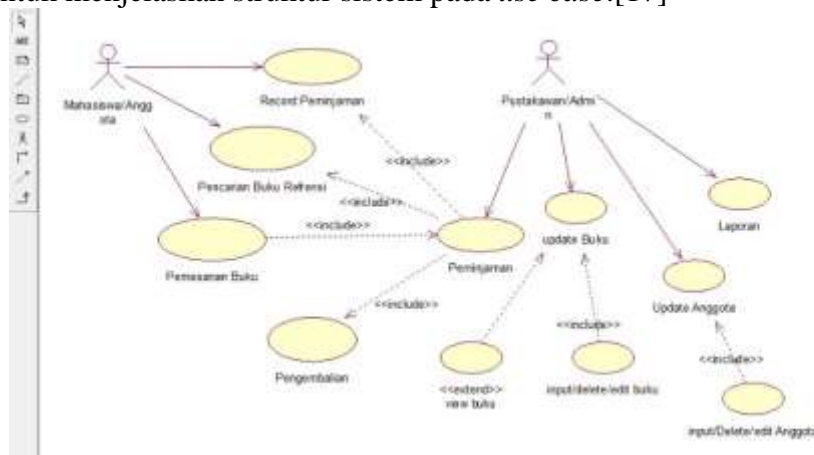


Gambar 5. Desain Arsitektur Buku Referensi

3. Hasil Penelitian Dan Pembahasan

3.1. Use Case Diagram

Setiap langkah dalam use case adalah sebuah elemen dalam interaksi antara actor dan sistem. Setiap langkah harus berupa pernyataan sederhana dan dengan jelas menunjukkan siapa yang menjalankan langkah tersebut. *Use case* adalah abstraksi dari interaksi antara system dan aktor. Analisis dengan menggunakan model *use case*, memberikan keleluasaan antara peneliti dan user untuk dapat menganalisa area permasalahan, sehingga interaksi antara user dengan sistem dapat dianalisa secara keseluruhan selama proses pengembangan sistem. Syarat-syarat yang diperlukan dalam pembuatan model *use case* adalah aktor, *use case* serta komunikasi hubungan antara aktor dengan *use case*. Penggambaran *use case* dengan *relationshipnya* dengan aktor membantu untuk menjelaskan struktur sistem pada *use case*. [17]



Gambar 7. Use Case Diagram

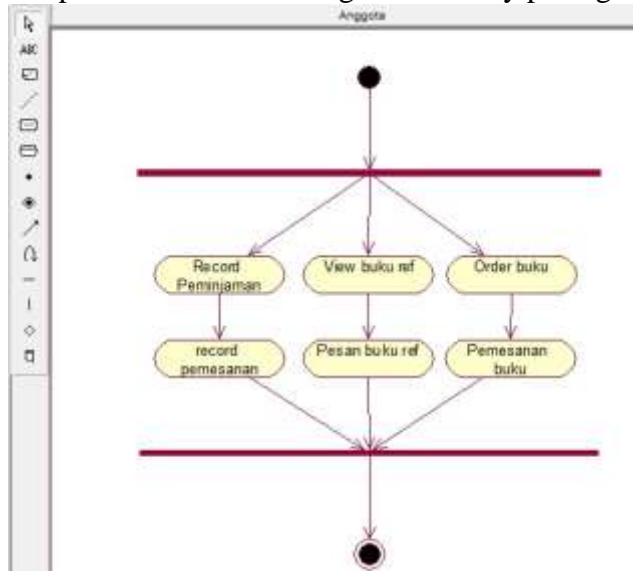
Tabel.1 Diskripsi Pengguna Use Case

No	Pengguna	Deskripsi
1	Mahasiswa/Anggota	Mencari buku/jurnal berdasarkan referensi yang di butuhkan.
2	Pustakawan	Melayani proses peminjaman dengan menginformasikan letak dengan ada atau tidaknya buku/junal yang memiliki referensi yang sesuai dengan kebutuhan

3.2. Activity Diagram

Activity diagram merupakan *state* diagram khusus, maka *activity* diagram tidak menggambarkan *behaviour* internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. *Activity* diagram merupakan gabungan dari beberapa teknik, yaitu: *event* diagrams oleh Jim Odell, *SDL* state modeling teknik, *work-flow modeling*, dan petri nets. *Activity* diagram menggambarkan berbagai alir aktivitas (*workflows*) dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.[18] Alur diagram aktifitas dan dalam alur informasi untuk anggota dan admin. Dapat dilihat melalui diagram *Activity* pada gambar 8 dan 9.

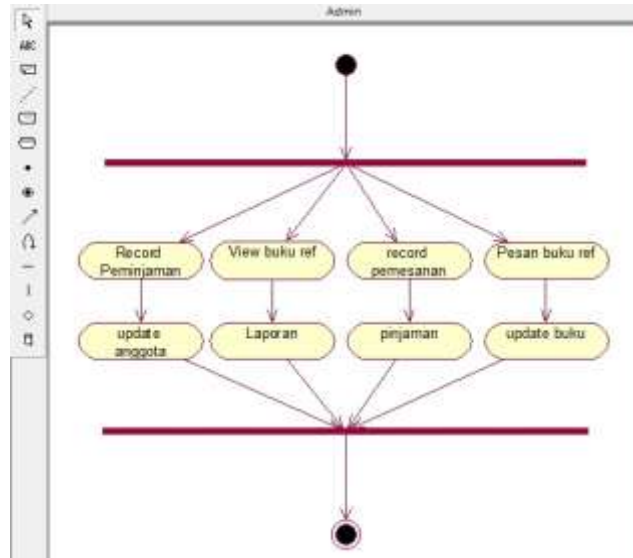
Berikut alur diagram aktifitas dan dalam alur informasi untuk anggota dengan attribut record peminjaman, view buku referensi, dan order buku. Dapat dilihat melalui diagram *Activity* pada gambar 8.



Gambar 8. Activity Diagram untuk Anggota

Berikut alur diagram aktifitas dan dalam alur informasi untuk admin dengan attribut Record peminjaman, View buku referensi, dan

Record Pesanan, Pesan buku Referensi. Dapat dilihat melalui diagram Aktivitas pada gambar 9.

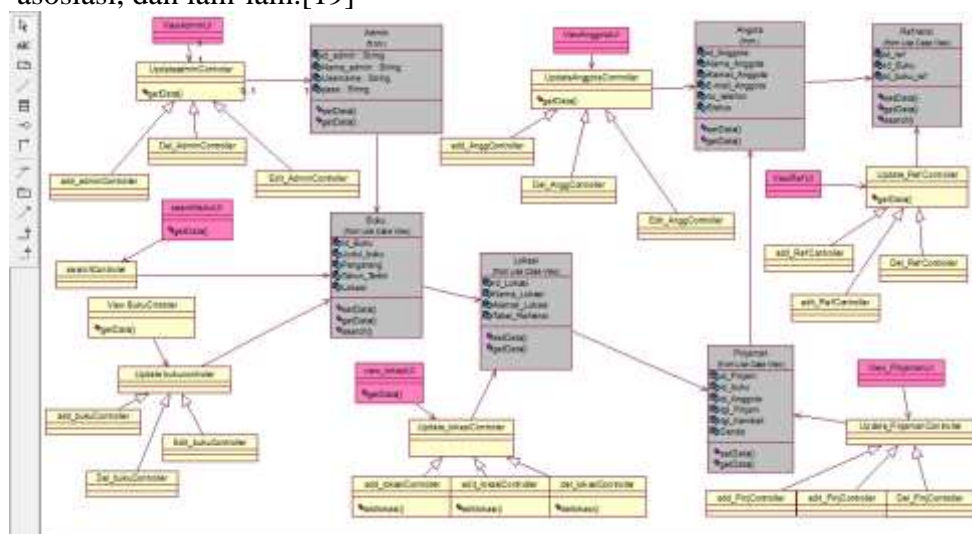


Gambar 9. Activity Diagram untuk Admin

3.3. Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (*atribut/property*) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (*metoda/fungsi*).

Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain.[19]



Gambar 10. Class Diagram Perpustakaan

3.5. Implementasi Restful

Membangun layanan dengan *web service* menggambarkan dari alur proses bisnis dari masing-masing bagian. Service yang dibangun berdasarkan parameter yang dibutuhkan dari masing-masing bagian yang langsung dengan software aplikasi lain dengan menggunakan message berbasis XML melalui protokol internet. Pesan yang disampaikan melalui XML/JASON akan ditampilkan oleh elemen REST respon dan request. Dalam pengkasesan *resource* REST juga menggunakan URL dimana ada method yang di gunakan, dengan default adalah *GET*, *PUT*, *DELETE*, *POST*.

```

@GET
@Override
@Produces({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
public List<Buku> findAll() {
    return super.findAll();
}

@GET
@Path("/{from}/{to}")
@Produces({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
public List<Buku> findRange(@PathParam("from") Integer from, @PathParam("to") Integer to) {
    return super.findRange(new int[]{from, to});
}

@POST
@Override
@Consumes({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
public void create(Buku entity) {
    super.create(entity);
}

@PUT
@Path("/{id}")
@Consumes({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
public void edit(@PathParam("id") String id, Buku entity) {
    super.edit(entity);
}

@DELETE
@Path("/{id}")
public void remove(@PathParam("id") String id) {
    super.remove(super.find(id));
}

@GET
@Path("/{id}")
@Produces({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
public Buku find(@PathParam("id") String id) {
    return super.find(id);
}

```

Gambar 11. Implementasi Restful

Hasil pengujian tahap simulasi dengan REST *web service*. Pada masing-masing bagian dapat diketahui dari proses inputan informasi dari tiap bagian yang dijadikan sebagai dasar untuk menentukan kebutuhan buku yang di jaiskan inputan, refreansi buku dan alamat lokasi.



Gambar 12. Header Request

Proses uji coba diberikan beberapa skenario untuk mengetahui fungsionalitas fungsionalitas dari program. Uji coba dilakukan mulai dari masuknya input dan diproses sampai akhirnya menghasilkan output. Uji coba meliputi pengujian beberapa komponen utama dari simulasi aplikasi ini. Dalam hasil test yang di berikan pada proses request dan respon nya

```
<?xml version="1.0" encoding="UTF-8"?>
<Perus url="http://localhost:8080/Perus/webresources/entitas.refrensi">
  <Perus url="http://localhost:8080/Perus/webresources/entitas.refrensi/1/">
    <idrefrensi>0005.43</idrefrensi>
    <idbukuref>S783o</idbukuref>
    <lokasi
  <Perus url="http://localhost:8080/Perus/webresources/entitas.lokasi/1/lokasi/">
    <idlokasi>AB347</idlokasi>
    <namalokasi>UPT</namalokasi>
    <alamatlokasi>repository lt 2</alamatlokasi>
  </refrensi>
</Perus url="http://localhost:8080/Perus/webresources/entitas.refrensi/2/">
```

Gambar 13. Header Response

4. Pembahasan

Berdasarkan analisis dan perancangan yang dilaksanakan pada bab-bab sebelumnya dihasilkan service-service yang berbasis RESTful yang terdapat pada server dan aplikasi informasi tiap bagian admin atau pustakawan sebagai client pengguna service-service ini. Empat service utama dan method yang telah dihasilkan pada bab sebelumnya kemudian akan diuji dan diintegrasikan. Berikut hasil implementasi service-service tersebut:

2. @GET

@Get service ini berfungsi untuk mengambil resource yang hanya perlu di baca saja (*Read Only*) service list buku dari sisi server. Terdapat 2 *method* yang diimplementasikan yaitu method: Menampilkan list Buku. {findRange (id)} dan method : menampilkan data/resource. (getScheduleInfo)

3. @POST

@POST service ini berfungsi untuk mengupdate suatu resource dari kebutuhan yang harus di gunakan sebagai data inputan ke sisi server. Terdapat 2 *method* yang diimplementasikan yaitu method: Mengupdate data/resource {create (entity)} dan method: Mengupdate data/resource. (getScheduleInfo)

4. @PUT

@PUT berfungsi untuk membuat atau create resource baru untuk kebutuhan yang harus di gunakan sebagai data inputan ke sisi server. Terdapat 2 *method* yang diimplementasikan yaitu method: Membuat data/resource baru {edit (entity)} dan *method*: Memebuat data/resource baru. (getScheduleInfo)

5. @DELETE

@PUT berfungsi untuk menghapus resource untuk kebutuhan yang tidak lagi harus di gunakan sebagai data inputan ke sisi server. Terdapat 2 *method* yang diimplementasikan yaitu method: Menghapus data/resource {edit (entity)} dan method: Menghapus data/resource. (getScheduleInfo)

Dari hasil *web service* yang di bangun dalam Perancangan Arsitektrur RESTful untuk integrasi data dari satu *web service* yang berisi banyak operation. Berdasarkan hasil yang telah dilakukan pada uji coba dan sudah di hasilkan berdasarkan implementasi. 1) Tiap bagian mendapatkan informasi data yang di butuhkan baik dari anggota dan admin sesuai dengan kebutuhannya berdasarkan *web service*, sehingga tidak lagi mendapat informasi data buku saja melainkan mengetahui list referensi buku dan di lokasi buku referensi berada. 2) Bagian Admin dapat memberikan informasi dan memiliki data lokasi buku referensi baik di perpustakaan PT atau perustakaan daerah.

5. Kesimpulan dan Saran

Pada penelitian ini menyimpulkan beberapa hal dengan hasil yang telah dilakukan. Integrasi beberapa *services* yang berasal dari membangun komunikasi dan mendistribusikan data atau informasi antar sistem merupakan langkah yang dapat ambil untuk mengatasi permasalahan keterbatasan informasi. Langkah tersebut dapat dibuktikan dengan cara pembentukan arsitektur integrasi, penyediaan laman, ketersediaan informasi baik referensi buku yang berkaitan dan lokasi buku sesuai referensi buku yang di harapkan, proses yang memberikan kemampuan integrasi antar sistem yang dapat digunakan dalam menunjang setiap aktifitas dalam lingkungan kelompok maupun individu. RESTful WS merupakan suatu cara yang dapat digunakan dalam mengintegrasikan sistem dan mendistribusikan data untuk sistem yang berbeda-beda. Dalam hal pengamanan sumberdaya, langkah yang dapat diambil yaitu dengan cara menggabungkan RESTful.

Saran penulis antara lain: 1. Dalam melakukan analisa dan perancangan menggunakan prinsip-prinsip RESTful agar benar-benar memperhatikan proses

bisnis yang ada supaya dihasilkan service yang dapat benar-benar independent atau tingkat kebergantungan antar service sangat kecil. 2. Service yang dihasilkan dalam penelitian ini dapat dikembangkan dengan perancangan dengan REST tahap lanjut untuk mengembangkan alamat lokasi buku referensi yang ada di fakultas – fakultas di beberapa universitas.

DAFTAR PUSTAKA

- [1] E. Yudhistira, A. Purwinarko, and I. U. Wusqo, “Implementasi Restful Web Service Menggunakan AsyncTask pada Aplikasi Library Automation Berbasis Android,” no. Snik, pp. 286–292, 2016.
- [2] E. Christianto, W. Utomo, and W. Sulistyono, “Data integration of cost materials needs through a service oriented architecture (study case: pt x garment ungaran),” *J. Adv. Comput. Sci. Technol.*, vol. 4, no. 1, pp. 136–147, 2015.
- [3] A. Wijanarko, I. Wisnubhadra, and B. L. Sinaga, “Integrasi informasi penelitian pada perpustakaan perguruan tinggi berbasis web service,” pp. 7–12, 2013.
- [4] Ramdani, “Pemanfaatan Koleksi Referensi Sebagai Sumber Daya Perpustakaan Dalam Jasa Layanan Informasi: Studi Kasus Pada Perpustakaan Mahkamah Konstitusi Republik Indonesia,” 2010.
- [5] A. I. Mujab and A. Setyadi, “Persepsi Pemustaka Terhadap Sikap Pustakawan dalam Layanan Referensi di Perpustakaan Universitas Katolik Soegijapranata,” vol. 4, no. 2, 2015.
- [6] E. Newcomer, *Understanding Web Services: XML, WSDL, SOAP, and UDDI*. 2004.
- [7] T. Erl, *Service-Oriented Architecture: a field guide to integrating XML and Web services*. 2004.
- [8] K. Laudon and J. P. Laudon, *Management Information Systems, Global Edition*. 2013.
- [9] R. S. Pressman and B. R. Maxim, *Software Engineering: a Practitioner’S Approach*. 2015.
- [10] F. Caeldries, “Reengineering the Corporation: A Manifesto for Business Revolution.,” *Acad. Manag. Rev.*, vol. 19, no. 3, pp. 595–600, 1994.
- [11] I. Dallas and M. T. Wynn, “Information Systems for Small and Medium-sized Enterprises,” pp. 25–47, 2014.
- [12] P. Sousa, C. Pereira, R. Vendeirinho, A. Caetano, and J. Tribolet, “Applying the Zachman Framework Dimensions to Support Business Process Modeling,” *Digit. Enterp. Technol.*, pp. 359–366, 2005.
- [13] J. Sandoval, *RESTful Java Web Services: Master Core REST Concepts and Create RESTful Web Services in Java*, vol. 11. 2009.
- [14] P. F. Tanaem, “RESTful Web Service Untuk Sistem Pencatatan,” vol. 2, no. April, pp. 2443–2229, 2016.
- [15] C. Pautasso, O. Zimmermann, and F. Leymann, “Restful web services vs. ‘big’ web services: making the right architectural decision,” *Proceeding 17th Int. Conf. World Wide Web*, pp. 805–814, 2008.
- [16] HostBridge Technology, “Choosing formal and informal Web services

- for,” in *SOAP and REST: Choosing formal and informal Web services for CICS integration*, no. 866, 2009.
- [17] M. Resiko, S. Kasus, and D. I. Pt, “DESAIN APLIKASI PENGENDALIAN KREDIT DAN BUMIPUTERA INDONESIA TBK,” 2008.
- [18] N. Sopiah, “Penggunaan Metode Analisis Dan Rancangan Berorientasi Objek Pada Web Jurnal Ilmiah Terpadu,” *semnasIF 2012*, vol. 2012, no. semnasIF, pp. 188–195, 2012.
- [19] I. Zufria, S. Utara, M. Indonesia, I. T. Quality, and T. M. Decisions, “Pemodelan Berbasis UML (Unified Modeling Language) dengan Strategi Teknik Orientasi Objek User Centered Design (UCD ...,” no. January 2013, 2016.