

## **Implementasi *Service Oriented Architecture* Menggunakan *Web Service* pada Sistem Informasi Donasi di Badan Wakaf Alquran**

**Yan Sofyan A.S**

Prodi SISTEM Komputer STMIK Indonesia, Jakarta  
e-mail: yansofyan@gmail.com

**Abstrak** – *Badan Wakaf Alquran adalah lembaga nirlaba yang bergerak dalam bidang wakaf, infak, zakat dan dana sosial lainnya. Donasi berasal dari perorangan, lembaga atau perusahaan. Dalam mengelola donasi dan melaksanakan program atau proyek sosial, Badan Wakaf Alquran menggunakan sistem informasi donasi berupa aplikasi dari beragam platform dan vendor. Perbedaan aplikasi dan platform mengakibatkan masalah pada integrasi data, khususnya perbedaan data donasi dan donatur pada beragam aplikasi tersebut. Solusi yang dapat digunakan adalah dengan mengembangkan model sistem informasi yang dapat mengintegrasikan data dari beragam aplikasi. Penelitian ini menerapkan Service Oriented Architecture (SOA) pada pengembangan sistem informasi donasi berupa web service yang bisa berkomunikasi atau diakses oleh beragam aplikasi. Web service merupakan aplikasi yang dapat menyediakan data maupun fungsi tertentu bagi aplikasi lain meskipun berbeda sistem operasi, perangkat keras, maupun bahasa pemrograman yang digunakan untuk membangunnya. Penelitian ini menghasilkan sebuah prototipe web service yaitu yang berfungsi sebagai media pertukaran atau sinkronisasi data donasi dan data donatur pada berbagai aplikasi yang dimiliki oleh Badan Wakaf Alquran.*

**Kata Kunci:** *web service, service oriented architecture, sistem informasi donasi*

**I.****PENDAHULUAN**

Sementara eksekutif TI telah menghadapi tantangan untuk memotong biaya dan memaksimalkan pemanfaatan teknologi yang ada, pada saat yang sama mereka harus terus berusaha untuk melayani pelanggan dengan lebih baik, lebih kompetitif, dan lebih responsif terhadap prioritas strategis bisnis..

Ada dua tema mendasar di balik semua tekanan ini: heterogenitas dan berubah. Sebagian besar perusahaan saat ini mengandung berbagai sistem yang berbeda, aplikasi, dan arsitektur dari berbagai usia dan teknologi. Mengintegrasikan produk dari beberapa vendor dan berbagai platform hampir selalu mimpi buruk Tapi kita juga tidak bisa mengambil pendekatan vendor tunggal ke IT, karena suite aplikasi dan infrastruktur pendukungnya begitu tidak fleksibel.

Perubahan adalah tema kedua yang mendasari pertanyaan eksekutif TI saat ini menghadapi. Globalisasi dan e-bisnis mempercepat laju perubahan.

Globalisasi menyebabkan persaingan sengit, yang menyebabkan produk menjadi lebih pendek siklus, karena perusahaan mencari keuntungan dari persaingan mereka. Pelanggan kebutuhan dan persyaratan berubah lebih cepat didorong oleh penawaran kompetitif dan kekayaan informasi produk yang tersedia melalui Internet. Sebagai tanggapan Siklus perbaikan kompetitif dalam produk dan layanan semakin meningkat. Perbaikan teknologi terus berakselerasi, memberi makan laju yang meningkat mengubah kebutuhan pelanggan Bisnis harus cepat beradaptasi untuk bertahan hidup, ayo sendiri untuk berhasil dalam lingkungan persaingan dinamis saat ini, dan TI infrastruktur harus memungkinkan kemampuan bisnis untuk beradaptasi.

**II. LANDASAN TEORI**

Memuat teori-teori pendukung dari metode yang diusulkan untuk pemecahan suatu masalah dan/atau pengembangan dari metode tersebut, yang didasarkan referensi yang jelas (buku, jurnal, prosiding dan artikel ilmiah lainnya).

Widya, M.A. dan Sensuse, D.I. (2017) dalam penelitiannya membangun model *mobile computing* untuk meningkatkan layanan e-government di tingkat desa berbasis SOA. Somantri, Oman dan Hasta, I.D. (2017). dalam penelitiannya menggunakan pendekatan Service Oriented Architecture (SOA) untuk mengoptimalkan sistem yang dibangun sehingga mempunyai peluang untuk dikembangkan lebih jauh lagi khususnya pada model web service untuk optimalisasi e-government pada Kelurahan Pesurungan Lor kota Tegal. Hodijah, Ade (2017), dalam penelitiannya merancang model pengembangan perangkat lunak berbasis *Service Oriented Architecture* menggunakan metodologi *Unified Process*. Dengan metode ini pada analisis proses bisnis saat ini (*as-is*) maka akan teridentifikasi rancangan kandidat-kandidat *service* sesuai model *service (to-be)* yang akan dikembangkan. Afwani, Royana. (2016) dalam penelitiannya melakukan implementasi Service Oriented Architecture (SOA) pada sistem informasi bergerak

pengobatan *tuberculosis* di Nusa Tenggara Barat. Melalui pengimplementasian SOA dihasilkan sistem informasi bergerak pengobatan TB di NTB yang fleksibel dengan serangkaian *service* yang *reusable* dan dapat digunakan pada berbagai platform perangkat bergerak.

**2.1. Service Oriented Architecture**

Menurut Sprott & Wilkes (2004) *Service oriented architecture* atau SOA didefinisikan sebagai kebijakan, praktek dan kerangka kerja yang memungkinkan fungsionalitas aplikasi disediakan dan dikonsumsi sebagai seperangkat *service* pada sebuah unit yang sesuai dengan kebutuhan *service* pemakai. *Service* dapat digunakan, dipublikasikan, ditemukan, dan diabstraksikan menggunakan bentuk antarmuka yang telah distandarkan.

Sedangkan menurut Bieberstein (2008) SOA adalah sebuah kerangka kerja untuk memadukan proses bisnis dan infrastruktur teknologi informasi yang aman dan dalam komponen terstandarisasi (*service*) yang dapat digunakan kembali (*reused*) dan dikombinasikan untuk mencapai perubahan prioritas bisnis.

Dengan definisi diatas, kita bisa menyimpulkan SOA adalah suatu cara perancangan aplikasi dengan menggunakan komponen-komponen yang sudah ada. Dan satu hal yang penting bagi SOA adalah komponen atau *service* tersebut harus bersifat *loose coupled* (tingkat ketergantungan antar komponen rendah). Sebuah *service* bisa dipanggil oleh aplikasi atau *service* lainnya tanpa melihat platform apa yang digunakan oleh *service* tersebut dan dimana lokasi *service* tersebut berada. Diharapkan dengan infrastruktur teknologi seperti ini, memungkinkan kemampuan bisnis untuk beradaptasi lebih cepat.

**2.2. Web Service**

*Web service* adalah sebuah sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas interaksi mesin-ke-mesin melalui sebuah jaringan (W3C, 2004). *Web service* secara teknis memiliki mekanisme interaksi antar sistem sebagai penunjang interoperabilitas, baik berupa agregasi (pengumpulan) maupun sindikasi (penyatuan). *Web service* 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.

Sekalipun mirip dengan Application Programming Interface (API) berbasis web, *web service* lebih unggul karena dapat dipanggil dari jarak jauh melalui internet. Pemanggilan *web service* bisa menggunakan bahasa pemrograman apa saja dan dalam platform apa saja, sementara API hanya bisa digunakan dalam platform tertentu [2]. *Web service* dapat dipahami sebagai Remote Procedure Call (RPC) yang mampu memproses fungsi-fungsi yang didefinisikan pada sebuah aplikasi web dan mengekspos sebuah API atau User Interface (UI) melalui web. Kelebihan *web service* adalah: 1) lintas platform, 2) language independent, 3)

jembatan penghubung dengan database tanpa perlu driver database dan tidak harus mengetahui jenis DBMS, 4) mempermudah proses pertukaran data, serta 5) penggunaan kembali komponen aplikasi [2]

**2.3. REST Web Service**

Cara kerjanya, REST server menyediakan jalur untuk akses resource atau data, sedangkan REST client melakukan akses resource dan kemudian menampilkan atau menggunakannya. Resource yang dihasilkan sebenarnya berupa teks, namun formatnya bisa bermacam-macam tergantung keinginan developer, umumnya adalah JSON dan XML. Dalam mengakses sebuah resource, REST juga menggunakan konsep URI dimana ada method yang digunakan adalah GET. Berikut ini method-method yang mendukung REST: 1) GET, cocok untuk resource yang hanya perlu dibaca saja (read only). 2) PUT, cocok digunakan untuk membuat/create resource baru. 3) DELETE, cocok digunakan untuk menghapus suatu resource. 4) POST, cocok digunakan untuk mengupdate suatu resource. 5) OPTIONS, cocok digunakan untuk mendapatkan operasi yang di support pada resource.

**III. METODE PENELITIAN**

**3.1. Tempat dan Waktu Penelitian.**

Tempat penelitian dilakukan di Badan Wakaf Al-Qur'an pada bagian Fundraising dan Keuangan.

Waktu penelitian dilakukan selama 2 bulan, yaitu bulan Agustus dan September.

Adapun yang diteliti pada bagian ini yaitu proses bisnis yang berlangsung, data dan informasi donasi atau wakaf, pihak-pihak yang terlibat dalam bidang ini serta hubungan bidang ini dengan bidang lainnya.

**3.2. Perangkat Keras dan Perangkat Lunak**

Dalam penelitian ini penulis menggunakan perangkat berikut ;

1. Komputer/Laptop.
2. Windows 10
2. Internet Information Server sebagai web server.
3. Bahasa Pemrograman C#
4. Teknologi API dan Framework: ASP.Net, Web API dan MVC
5. DBMS Microsoft SQL Server
6. IDE Visual Studio 2019.

**3.3. Prosedur Penelitian.**

Prosedur yang dilakukan dalam pembuatan penelitian ini adalah sebagai berikut:

1. Pengumpulan Data.
  - a. Data Primer.

Pengumpulan data dilakukan secara langsung melalui pengamatan dan wawancara di lapangan dan wawancara langsung jika dibutuhkan. Kegiatan ini dilakukan untuk mengidentifikasi masalah yang ada sehingga penulis dapat mengetahui apa yang dibutuhkan dalam penyusunan penelitian

- ini.
- b. Data sekunder.

Untuk melengkapi dan menyempurnakan penelitian ini, penulis menyusun landasan teori yang berkesesuaian berdasarkan buku-buku, jurnal maupun makalah yang berkaitan dengan penelitian ini. Penulis juga melakukan *browsing* beberapa situs internet untuk mengumpulkan informasi yang berhubungan dengan hal-hal yang dibahas pada penelitian ini.

**2. Analisis dan Perancangan Sistem.**

Berdasarkan data yang diperoleh, dilakukan proses analisis terhadap sistem yang berjalan dan perancangan berdasarkan pada prinsip-prinsip *Service Oriented Architecture*.

**3. Implementasi dan Pengujian Sistem.**

Rancangan Web Services yang dihasilkan sesuai prinsip-prinsip SOA, Kemudian diuji menggunakan aplikasi Web Service atau Web API client. Pengujian dilakukan untuk mengetahui kesesuaian informasi yang dihasilkan dan informasi yang diharapkan.

**IV. PEMBAHASAN**

Berdasarkan pengamatan dan wawancara yang dilakukan, didapatkan gambaran fungsi dan sistem yang berjalan saat ini. Lihat tabel 1, berikut ini:

**Tabel 1.**  
**Sistem Informasi Yang Berjalan**

No	Fungsi	Sistem Informasi
1	Pengelolaan Keuangan dan Akuntansi	Sistem Informasi Keuangan dan Akuntansi (inovaGL)
2	Pengelolaan Donasi ZISWAF dan Data Donatur/Wakif	Sistem Informasi Manajemen Donasi (MgmDonasi)
3	Pengelolaan Informasi dan Donasi melalui Web	Web Resmi Organisasi dan Sistem Informasi Donasi Online (WEB)
4	Pengelolaan Aktivitas Telemarketing	Sistem Informasi Relasi Pelanggan (CRM)
5	Pengelolaan Manajemen Karyawan	Sistem Informasi Administrasi Karyawan (SIKAK)
6	Pengelolaan Laporan Manajemen	Sistem Informasi Eksekutif (SIE)

Tabel 1. memberikan gambaran sistem informasi/aplikasi yang digunakan dalam pengelolaan donasi di bagian fundraising dan keuangan/akuntansi.. Hubungan keterkaitan sistem informasi yang berjalan dengan proses kerja menggambarkan bagaimana dukungan aplikasi/sistem informasi yang ada terhadap

fungsi kerja pada bagian terkait. Dari hasil pemetaan fungsi dan proses kerja terhadap aplikasi yang ada, didapatkan beberapa kandidat layanan-layanan yang terkait dengan fungsi donasi, kandidat layanan-layanan tersebut seperti yang terlihat pada tabel II berikut ini:

Tabel 2.

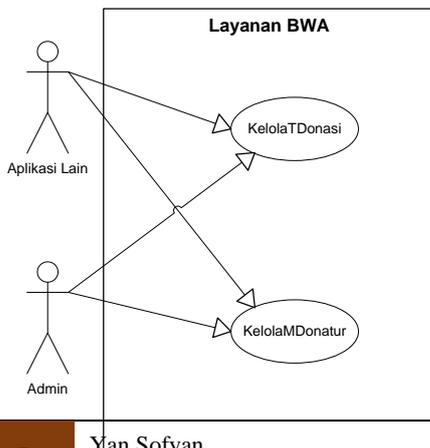
**Matrik Hubungan Proses Kerja Dengan Aplikasi Berjalan Yang Membentuk Kandidat Layanan**

No	Proses Kerja	Aplikasi yang Ada	Kandidat Layanan
1	Penerimaan Donasi ZISWAF	MgmDonasi, inovaGL, CRM, WEB	Layanan KelolaTDonasi
2	Pelaksanaan Administrasi Donatur	MgmDonasi, WEB, CRM	Layanan KelolaMDonatur
3	Pelaksanaan Administrasi Karyawan	MgmDonasi, SIAK, CRM, RS	Layanan KelolaMKaryawan
4	Pelaksanaan pembuatan laporan eksekutif	MgmDonasi, inovaGL, CRM, WEB, SIAK, RS	Layanan KelolaEksekutif

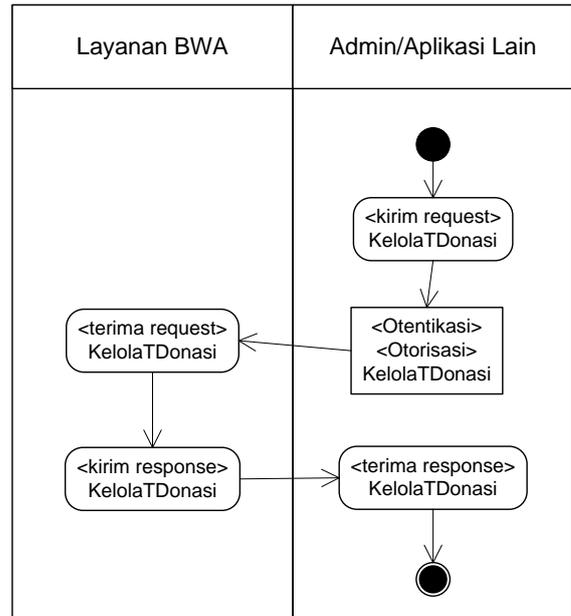
**4.1. Model Arsitektur Integrasi Sistem**

Penyusunan model arsitektur sistem terintegrasi berdasarkan hasil dari analisis identifikasi kebutuhan layanan yang diprioritaskan oleh pengguna.. Model arsitektur sistem terintegrasi dengan pendekatan SOA yang di rancang, berhubungan dengan beberapa aplikasi hasil analisis kebutuhan pengguna pada bagian fundraising dan keuangan/akuntansi.. Tabel 2. menjelaskan terdapat 4 (empat) proses kerja yang memiliki hubungan dengan aplikasi yang ada. Dari empat proses kerja tersebut diprioritaskan menjadi hanya 2 (dua) kandidat layanan yaitu layanan Tdonasi dan layanan MDonatur.

Gambar 1. Use Case Layanan



Gambar 2. Activity Diagram



Gambar 3. Sequence diagram



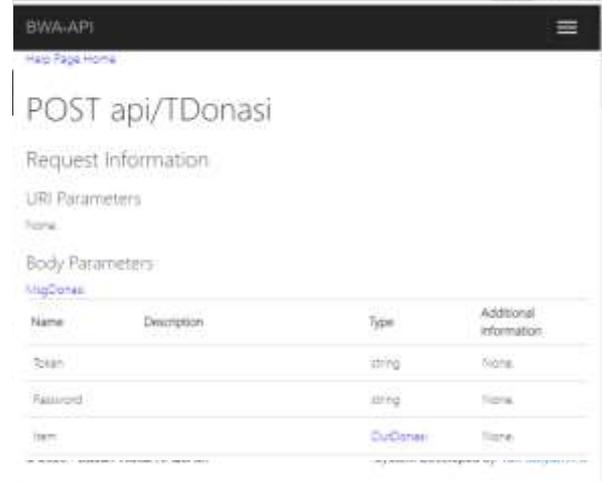
```

{
  "Token": "sample string 1",
  "Password": "sample string 2",
  "Item": {
    "ID": 1,
    "NoKwitansi": "sample string 2",
    "TglTransaksi": "2020-09-30",
    "KdKas": "sample string 4",
    "KdDonatur": "sample string 5",
    "NmWakif": "sample string 6",
    "Total": 7.0,
    "Donatur": {
      "ID": 1,
      "KdDonatur": "sample string 2",
      "NmLengkap": "sample string 3",
      "Alamat": "sample string 4",
      "Kota": "sample string 5",
      "Pos": "sample string 6",
      "Propinsi": "sample string 7",
      "Telp": "sample string 8",
      "Hp": "sample string 9",
      "Email": "sample string 10"
    },
    "Items": [
      {
        "DtID": 1,
        "NoKwitansi": "sample string 2",
        "KdProgram": "sample string 3",
        "KdProject": "sample string 4",
        "Qty": 5.0,
        "Jmh": 6.0
      },
      {
        "DtID": 1,
        "NoKwitansi": "sample string 2",
        "KdProgram": "sample string 3",
        "KdProject": "sample string 4",
        "Qty": 5.0,
        "Jmh": 6.0
      }
    ]
  }
}
  
```

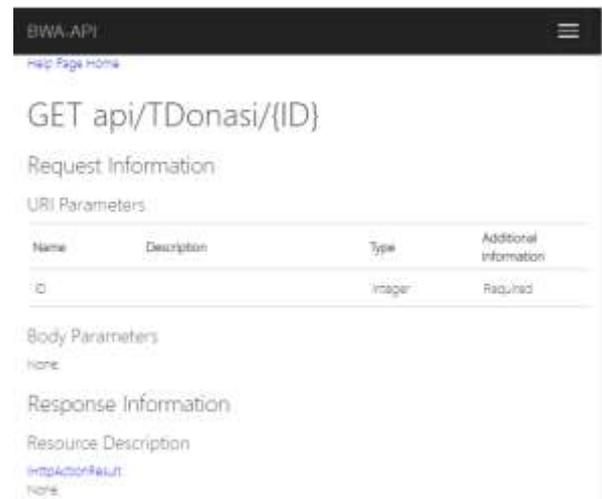
#### 4.2. Pembuatan Prototipe Web Service

Proses perancangan web service dibangun menggunakan teknologi Web-API dan MVC menggunakan bahasa C# dan ASP.NET pada framework Microsoft dot NET. Terdapat dua web method yang akan dibuat sebagai prototipe layanan web service, yaitu TDonasi dan MDonatur.

**Gambar 4. Halaman depan dokumentasi Web-API.**



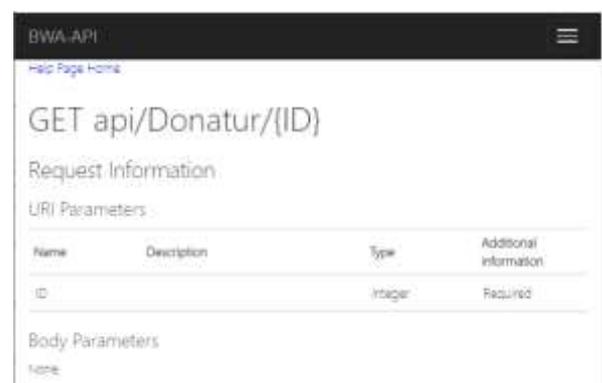
**Gambar 5. Melihat Transaksi Donasi berdasarkan ID**



**Gambar 6.. Tambah/Update Transaksi Donasi**

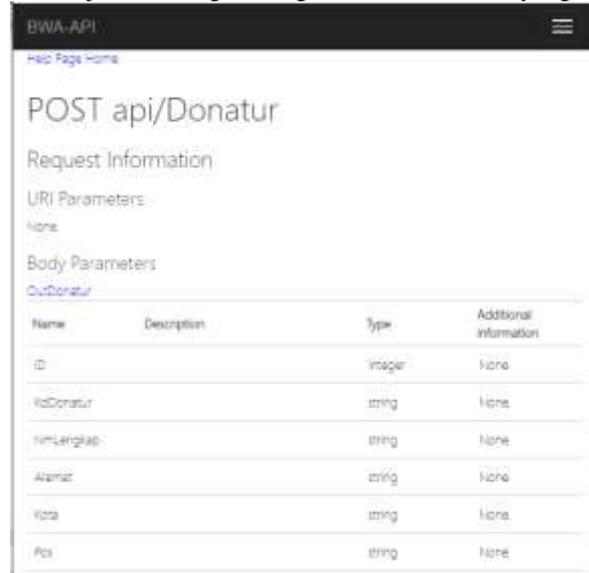
**Gambar 7. Request POST TDonasi dalam Format JSON**

**Gambar 8. . Melihat Donatur berdasarkan ID**



**Gambar 9. Tambah/Update Donatur**

terjadi. Masing-masing service dan modul yang ada pada



**Gambar 10. Request POST Donatur dalam Format JSON:**

```
{
  "ID": 1,
  "KdDonatur": "sample string 2",
  "NmLengkap": "sample string 3",
  "Alamat": "sample string 4",
  "Kota": "sample string 5",
  "Pos": "sample string 6",
  "Propinsi": "sample string 7",
  "Telp": "sample string 8",
  "Hp": "sample string 9",
  "Email": "sample string 10"
}
```

aplikasi memiliki skenario test, dimana terdapat test case yang beragam, kemudian akan diamati hasilnya apakah sesuai dengan yang dikehendaki. Skenario pengujian dengan pendekatan Black Box dilakukan pada aplikasi dari sisi server dan sisi client.

**Pengujian Dengan Metode Black Box Pada Layanan BWA**

**4.3. Pengujian**

Pengujian merupakan bagian yang sangat penting dalam membangun suatu perangkat/prototipe, pengujian ditujukan untuk menemukan kesalahan-kesalahan pada suatu sistem dan memastikan sistem yang telah dibangun sudah sesuai dengan apa yang direncanakan. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari suatu perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal yaitu mampu mempresentasikan kajian pokok dan spesifikasi analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri. Pengujian dilakukan untuk mengetahui apakah penerapan SOA dan metode MVC dapat menunjang web service yang sesuai dengan analisis kebutuhan yang telah ditentukan. Pengujian ke responden dilakukan oleh beberapa staf dilingkungan bagian fundraising dan keuangan/akuntansi fungsi penerimaan donasi dan pengelolaan donatur.. Pelaksanaan pengujian ini berdasarkan skenario yang telah ditentukan dengan setiap proses dan kemungkinan kesalahan yang dapat

ResponseTDonasi
ItemID
Status
Message
KdDonatur

ResponseMDonatur
ItemID
Status
Message

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Aplikasi mengirim request TDonasi (Format JSON)	Layanan mengirim balik response sesuai <i>class Response</i> yang telah dibuat dalam format JSON.	Response (JSON) sesuai dengan <i>class</i> yang ditentukan

2	Aplikasi mengirim request MDonatur (Format JSON)	Layanan mengirim balik response sesuai <i>class Response</i> yang telah dibuat dalam format JSON.	Response (JSON) sesuai dengan <i>class</i> yang ditentukan
---	--	---	--

Berdasarkan hasil pengujian *Blackbox* tersebut dapat disimpulkan bahwa layanan BWA dari sisi server sudah sesuai dengan spesifikasi kebutuhan pengguna. Berdasarkan hasil analisis, desain arsitektur menggunakan pendekatan SOA dan metode Web API pada Badan Wakaf Alquran (BWA) dapat berfungsi dalam menyediakan kebutuhan integrasi dan pertukaran data sehingga dapat memecahkan permasalahan integrasi sistem antar aplikasi yang ada di BWA.

**V. KESIMPULAN**

Pada penelitian ini diperoleh hasil rancangan arsitektur dan prototipe *web service* yang disebut layanan BWA. Rancangan arsitektur dibangun dengan pendekatan SOA dan diimplementasikan dengan teknologi *Web API* dari Microsoft. Prototipe Layanan BWA yang dihasilkan adalah layanan transaksi donasi dan layanan donatur. Prototipe layanan yang dibuat hanya di sisi server berupa API yang bisa di-consume atau digunakan oleh berbagai aplikasi.

Dengan implementasi layanan BWA ini memungkinkan pertukaran dan sinkronisasi data antara aplikasi yang dimiliki BWA, khususnya data donasi dan donatur, sehingga mampu meminimalisir bahkan menghilangkan perbedaan data donasi dan donatur yang digunakan oleh berbagai aplikasi.

Selanjutnya hasil penelitian ini bisa diterapkan pada satuan kerja terkait di lingkungan Badan Wakaf Alquran (BWA) dan dikembangkan lebih lanjut mencakup layanan data lainnya yang dibutuhkan.

**REFERENSI**

Afwani, Royana. (2016). Implementasi Service Oriented Architecture (SOA) Pada Sistem Informasi Bergerak Pengobatan Tuberkulosis di Nusa Tenggara Barat. DIELEKTRIKA. 85 – 94.  
 Bieberstein, N., et.al. (2008). *Executing SOA: A Practical Guide for the Service Oriented Architect*. IBM Press.  
 Endrei, Mark,et.al. (2004). *Patterns: Service-Oriented Architecture and Web Services*. IBM Redbooks, 18-19.  
 Hodijah, Ade (2017). Perancangan Model Pengembangan Perangkat Lunak Berbasis SOA Menggunakan Metodologi Unified Process. Jurnal Penelitian dan Pengabdian Masyarakat, ETHOS. 253-261.  
 Hurwitz, Judith, et.al. (Tahun). *Service Oriented Architecture (SOA) For Dummies*. Nama Jurnal, Halaman.  
 Somantri, Oman dan Hasta, I.D. (2017). Jurnal

Pengembangan IT (JPIT) , Vol. 2, No. 1, Januari 2017. 23-29  
 Sprott D, & Wilkes L. 2004. *Understanding Service Oriented Architecture*. Microsoft Architect Journal.  
 Widya, M.A dan Sensuse, D.I. (2017). *Model Mobile Computing Untuk Meningkatkan Layanan E-Government Di Tingkat Desa*. Jurnal Sains dan Teknologi SAINTEKBU, Volume 9 No.2 Mei 2017, 7-15.  
 W3 Consortium, (2004). *Web Services Architectures*, diakses: 24-09-2017 dari <http://www.w3.org/TR/ws-arc>