

Pengembangan Sistem Monitoring Manajemen Yayasan Badan Wakaf Sultan Agung Berbasis REST API

Hisnan Faudan Azmia, Imam Much Ibnu Subroto, Suryani Alifah

Fakultas Teknolgi Industri, Prodi Teknik Informatika Universitas Islam Sultan Agung

Corresponding Author : hisnan@std.unissula.ac.id

Abstrak

Penelitian ini menerapkan teknologi *web services* dengan metode REST API pada Sistem Monitoring Manajemen Yayasan Badan Wakaf Sultan Agung untuk mengintegrasikan sistem yang berbeda platform dari unit yang berbeda sehingga kebutuhan informasi yang lengkap. Pengujian menunjukkan bahwa teknologi REST API berhasil merealisasikan interoperabilitas antara Sistem Monitoring Manajemen YBWSA dengan dua sistem lainnya yang berbeda platform. Pengujian REST API dilakukan dengan 2 cara yaitu dengan memasukan parameter sesuai dengan desain dari API Endpoint, dari pengujian tersebut REST API dapat mengembalikan data sesuai dengan parameter yang digunakan, kemudian jika data yang dicari maka akan mengembalikan HTTP *Status Code* '404' (*Not Found*) dan jika parameter tidak sesuai maka REST API mengembalikan data dalam kondisi default atau tanpa parameter, dan pengujian kecepatan dari pengujian yang dilakukan terjadi perbedaan ketika menggunakan parameter dengan tidak menggunakan parameter misalkan pada API Endpoint /unissula/penelitian yang dilakukan sebanyak 10 kali yang masing-masing dalam satuan milisecond (ms). Untuk rata-rata kecepatan tanpa menggunakan parameter adalah 106, 2 milisecond, sedangkan dengan menggunakan parameter "judul=learning&kdProdi=55201" adalah 80 milisecond. Kecepatan request dari REST API dipengaruhi beberapa hal yaitu, penggunaan *query*, ukuran dari database, optimalisasi database seperti penggunaan *query*, index dalam tabel dan kecepatan jaringan internet.

Kata Kunci : Interoperabilitas, REST API, Web Services

1. PENDAHULUAN

Yayasan Badan Wakaf Sultan Agung (YBWSA) merupakan Lembaga wakaf yang bergerak pada sektor wakaf dan dakwah, pendidikan, layanan kesehatan dan badan usaha pendukung dimana masing-masing dikelola oleh unit-unit dibawahnya. YBWSA perlu memantau kinerja unit-unit dibawahnya tersebut agar dapat bertahan dan berkembang, sehingga informasi yang lengkap, tepat dan cepat harus didapatkan dari unit-unit tersebut maupun sistem lainnya yang terkait. Untuk menunjang kegiatan pengolahan informasi setiap lembaga memiliki sistem informasi yang terpisah dengan platform yang berbeda-beda, sehingga data yang diolah sistem informasi tidak terpusat ke Yayasan Badan Wakaf Sultan Agung (YBWSA). Berdasarkan latar belakang diatas, maka didapatkan perumusan masalah yaitu, bagaimana agar dapat lembaga yang ada dibawah Yayasan dapat menyajikan data yang dibutuhkan berdasarkan sistem sudah ada kedalam sistem informasi monitoring manajemen YBWSA.

Penggunaan API (*Application Programing Interface*) menjadi salah satu solusi untuk mengatasi integrasi data dari platform yang berbeda. API adalah bagian antar muka dari program yang dapat dipanggil dan dijalankan oleh program lain. Interface yang dimaksud adalah sebuah *entry point* yang digunakan untuk mengakses sumber daya yang terdapat di *software* [1]. Misalkan pada bahasa pemrograman PHP, terdapat perintah *mysqli_connect* yang digunakan untuk melakukan koneksi ke *database management system* MySQL, kemudian pada javascript terdapat perintah *getElementById* yang digunakan untuk mendapatkan element dari HTML berdasarkan ID.

Dalam pertukaran data dengan internet, terdapat *web services* yaitu proses pertukaran pesan / informasi didalam jaringan, dengan menggunakan protokol HTTP / HTTPS (*Hyper Text Transfer Protocol / Secure*). Kemudian berdasarkan responnya *web services* dibagi menjadi 2 yaitu SOAP (*Simple Object Access Protocol*) dengan menggunakan XML (*Extensible Markup Language*) [2], dan REST (*Representational State Transfer*) menggunakan JSON (*Javascript Object Notation*) [3]. Dalam penelitian ini REST API sebagai metode pertukaran data untuk sistem monitoring manajemen untuk Yayasan Badan Wakaf Sultan Agung. REST API dipilih karena berbagai kelebihanannya penggunaan diantaranya JSON sebagai output dari data yang dikirim, karena JSON bisa diterima sebagian bahasa pemrograman saat ini, sedangkan SOAP yang menggunakan XML harus menggunakan parsing terlebih dahulu, kemudian penggunaan URI (*Uniform Resource Identifier*) untuk mengetahui mengidentifikasi resource berdasarkan nama, lokasi, atau keduanya [4], disertai HTTP *Method* yang digunakan seperti GET, POST, DELETE, PUT dan lainnya, kemudian dari request yang sudah dilakukan server akan menyertakan *http status code* untuk mengetahui request tersebut berhasil dilakukan atau gagal [5].

Terdapat beberapa penelitian terdahulu terkait bidang ini yaitu Dalam rancang bangun sistem informasi peringatan dini kejadian luar biasa di kota Pontianak, menjalankan model Interoperabilitas untuk mengatasi masalah dalam pengolahan data penduduk yang terdapat pada sistem informasi administrasi kependudukan dan data kesehatan dari Sistem Informasi Kesehatan di Kota Pontianak, dengan menggunakan metode REST API [6].

Untuk mempermudah pengumpulan data mahasiswa berprestasi di Fakultas Teknologi Pertanian Universitas Brawijaya, maka dibuat aplikasi berbasis web, dengan teknologi *web services* berbasis REST API yang dibangun dengan menggunakan 2 framework dengan bahasa pemrograman PHP yaitu laravel sebagai framework untuk aplikasi web-nya dan lumen sebagai REST API [7].

Masalah serupa dialami oleh Travel Tournic, karena sistem informasi yang dibangun memiliki fungsi-fungsi yang berbeda. Adapun solusi yang diberikan adalah melakukan pengembangan aplikasi dengan arsitektur *microservices* dengan menggunakan *web service* dengan metode REST API. pengembangan *web service* dengan metode REST API dibantu menggunakan bahasa pemrograman PHP dengan *framework* Laravel, node.js dan untuk REST API dibantu RAML (RESTful API Modeling Language), dengan bahasa YAML [8].

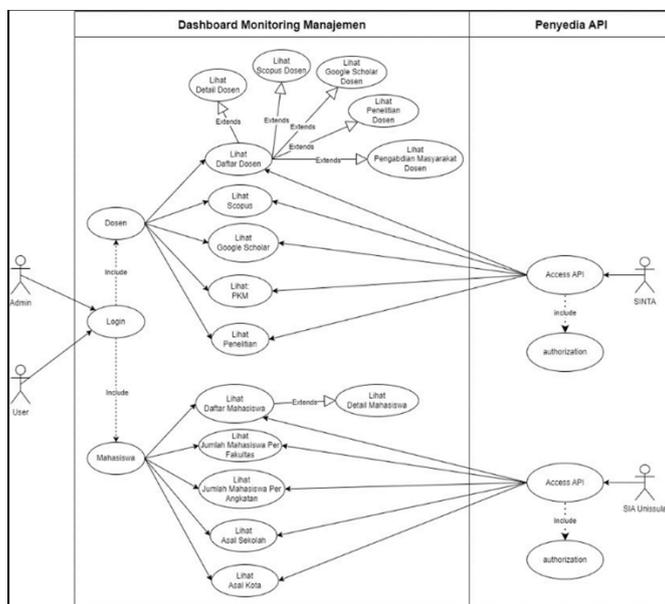
Dengan memanfaatkan REST API yang sudah dibuat oleh orang lain aplikasi Pemantau Coronavirus Disease 2019 (COVID-19) dapat dibuat tanpa memiliki sumber data sendiri, adapun REST API yang bersifat publik sehingga dapat digunakan oleh semua orang [9].

Adapun tujuan dari penelitian ini adalah merancang, merealisasikan, dan menguji kinerja dari sistem sistem monitoring manajemen Yayasan Badan Wakaf Sultan Agung memastikan agar data yang diterima sesuai dengan kebutuhan.

2. METODE PENELITIAN

2.1. Analisis Kebutuhan

Tahap ini bertujuan untuk mengidentifikasi platform dan karakteristik sistem informasi pada unit-unit yang berada di bawah YBWSA maupun fungsionalitas yang diharapkan. Pada penelitian ini dibatasi hanya dua sistem yang dieksplorasi dan akan diintegrasikan, yakni Sistem Informasi Terintegrasi (SIT) dari unit Pendidikan yaitu Universitas Islam Sultan Agung (Unissula), dan system terkait yakni sistem informasi *Science and Technology Index* (SINTA). Pada kasus ini akan ditunjukkan integrasi data mahasiswa dari SIA dan data publikasi dosen dari SINTA. Metode yang digunakan adalah dengan observasi langsung ke sistem terkait dan wawancara dengan pihak Unissula dan YBWSA. Selanjutnya dilakukan analisis sistem untuk mendapatkan gambaran umum dari sistem, seperti aktor, fungsionalitas sistem yang akan dimodelkan dengan *use case diagram*. Berikut ini adalah penggambaran dari use case yang ada terdapat pada gambar 1.

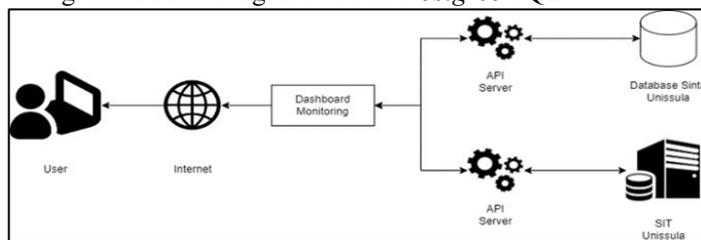


Gambar 1 Use Case Digram Sistem Monitoring Manajemen

2.2. Arsitektur Sistem

Gambar 2 menunjukkan arsitektur system yang terdiri dari dashboard monitoring, SIT Unissula, SINTA dan dua API Server masing-masing untuk memberikan layanan interoperabilitas dengan SIA Unissula dan SINTA. Kedua API Server di *install* Fusio API Management, yang berfungsi sebagai pengelola dan pemantuan dari REST API, dan untuk aplikasi dashboard monitoring dibangun menggunakan bahasa pemrograman PHP dengan framework Codeigniter. Data dosen akan diambil

dari SINTA yaitu portal jurnal untuk melihat kinerja penulis, peneliti dari perguruan tinggi yang dikembangkan dengan menggunakan bahasa pemrograman PHP dan database MySQL, sedangkan untuk data mahasiswa diambil dari SIA Unissula yaitu aplikasi web untuk melayani administrasi akademik di Universitas Islam Sultan Agung yang fiturnya disesuaikan dengan *role* / perannya, dimana platformnya menggunakan bahasa pemrograman PHP dengan database *Postgree SQL*.



Gambar 2 Arsitektur Sistem Monitoring Manajemen

2.3. Kandidat Web Services

Penentuan kandidat *web service* ditentukan berdasarkan fase analisis kebutuhan. Untuk data dosen yang diambil dari SINTA, ada ditabel 1, kemudian untuk data dari SIT Unissula ada ditabel 1.

Tabel 1. Kandidat web services dosen

No	Services	Deskripsi
1	Daftar Dosen	Menampilkan daftar dosen yang terdaftar di SINTA
2	Detail Dosen	Menampilkan data keseluruhan dari dosen
3	Scopus Dosen	Menampilkan jurnal yang masuk ke Scopus dari dosen
4	Scopus Universitas	Menampilkan jurnal yang masuk ke Scopus dari seluruh dosen
5	Google Scholar Dosen	Menampilkan jurnal yang masuk ke Google Scholar dari dosen
6	Google Scholar Universitas	Menampilkan jurnal yang masuk ke Google Scholar dari seluruh dosen
7	Penelitian Dosen	Menampilkan Penelitian yang sudah dilakukan oleh dosen
8	Penelitian Universitas	Menampilkan Penelitian yang sudah dilakukan oleh seluruh dosen
9	Pengabdian Masyarakat Dosen	Menampilkan Penelitian yang sudah dilakukan oleh dosen
10	Pengabdian Masyarakat Universitas	Menampilkan Penelitian yang sudah dilakukan oleh seluruh dosen

Tabel 1. Kandidat web services mahasiswa

No	Services	Deskripsi
1	Jumlah Mahasiswa Per Fakultas	Menampilkan jumlah mahasiswa per fakultas berdasarkan periode saat ini
2	Jumlah Mahasiswa Per Angkatan	Menampilkan jumlah mahasiswa per angkatan berdasarkan periode saat ini
3	Daftar Mahasiswa	Menampilkan daftar mahasiswa yang aktif
4	Detail Mahasiswa	Menampilkan detail dari mahasiswa
5	Asal Sekolah	Menampilkan daftar asal sekolah dari mahasiswa
6	Asal Kota	Menampilkan daftar asal kota dari mahasiswa

2.4. Identifikasi Web Services Interface

Web service interface berfungsi sebagai antarmuka dari *web service* yang akan berinteraksi dengan aplikasi klien. Karena menggunakan REST API, maka akan digunakan HTTP Method sebagai *interface*. HTTP Method yang digunakan ditentukan berdasarkan operasi yang dilakukan oleh *service* tersebut.

Tabel 3 Identifikasi Web Services Sinta Unissula

No	Services	HTTP Method
1	Daftar Dosen	GET
2	Detail Dosen	
3	Scopus Dosen	
4	Scopus Universitas	

5	Google Scholar Dosen	
6	Google Scholar Universitas	
7	Penelitian Dosen	
8	Penelitian Universitas	
9	Pengabdian Masyarakat Dosen	
10	Pengabdian Masyarakat Universitas	

Tabel 2 Identifikasi Web Services untuk Data Mahasiswa

No	Services	HTTP Method
1	Jumlah Mahasiswa Per Fakultas	GET
2	Jumlah Mahasiswa Per Angkatan	
3	Daftar Mahasiswa	
4	Detail Mahasiswa	
5	Asal Sekolah Mahasiswa	
6	Asal Kota Mahasiswa	

2.5. Perancangan REST API

Pada tahap ini dilakukan pendefinisian kontak REST API dengan menentukan desain URI (*Uniform Resource Identifier*) yang nantinya akan disebut sebagai *API Endpoint*. Desain URI bertujuan untuk membedakan pemanggilan setiap *resources* dari *services* agar bersifat unik, sehingga *client* dapat mengidentifikasi resource berdasarkan *services*-nya [10].

- a) Desain URI SINTA, Pada tabel 5 adalah desain URI untuk API Sinta Unissula, jika dibelakang parameter ada “:id” wajib memasukan id yang ada di dalam database sinta.

Tabel 5 Desain URI API Sinta Unissula

No	Services	Desain URI
1	Daftar dosen	/unissula/dosen
2	Detail Dosen	/unissula/dosen/:id
3	Scopus Dosen	/unissula/dosen/scopus/:id
4	Google Scholar	/unissula/dosen/google/:id
5	Penelitian Dosen	/unissula/dosen/penelitian/:id
6	Pengabdian Masyarakat	/unissula/dosen/pkm/:id
7	Scopus Universitas	/unissula/scopus
8	Google Scholar Universitas	/unissula/google
9	Penelitian Universitas	/unissula/penelitian
10	Pengabdian Masyarakat Universitas	/unissula/pkm

- b) Desain URI API Data Unissula, Pada tabel 6 adalah desain URI untuk API Unissula, jika dibelakang parameter ada “:nim” wajib memasukan nim yang ada di dalam database unissula.

Tabel 6 Desain URI API Unissula

No	Services	Desain URI
1	Jumlah mahasiswa per fakultas	/jumlah/fakultas
2	Jumlah mahasiswa per angkatan	/jumlah/angkatan
3	Daftar mahasiswa	/daftar/aktif
4	Detail Mahasiswa	/overview/:nim
5	Asal Sekolah	/asal/sekolah
6	Asal Kota	/asal/kota

3. HASIL DAN ANALISA

3.1. Pengujian Parameter

Untuk memastikan API Endpoint yang akan digunakan sesuai, maka diperlukan pengujian REST API yang dilakukan dengan metode Black box dengan bantuan aplikasi Postman. Blackbox dipilih karena pengujian dilakukan karena dapat memastikan setiap bagian dapat berjalan dengan baik atau tidak.

- 1) SINTA Unissula, pada tabel 7 adalah hasil pengujian penggunaan parameter pada API Endpoint “/unissula/dosen”

Tabel 7 Pengujian Parameter API Endpoint /unissula/dosen

Kasus Uji dengan parameter	Hasil Uji	Keterangan
Tanpa parameter	Menampilkan daftar dosen	Berhasil
kdFakultas='7'	Menampilkan daftar dosen dengan kode fakultas = 7	Berhasil
kdFakultas=7&item=5	Menampilkan daftar dosen dengan kode fakultas = 7 dan berada dihalaman 5	Berhasil
q=ali	Menampilkan daftar dosen tanpa filter	Berhasil
nama=firman	Menampilkan array kosong	Berhasil

- 2) Unissula, pada tabel 7 adalah hasil pengujian penggunaan parameter pada API Endpoint “/asal/sekolah”

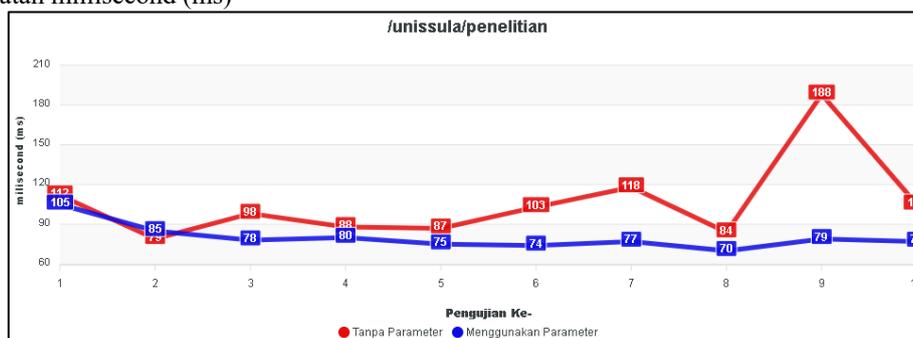
Tabel 8 Pengujian Parameter API Endpoint /asal/sekolah

Kasus Uji dengan parameter	Hasil Uji	Keterangan
Tanpa Parameter	Menampilkan asal sekolah dari mahasiswa	Berhasil
angkatan=2021	Menampilkan asal sekolah dari mahasiswa berdasarkan angkatan 2021	Berhasil
nama=sma&kota=pemalang	Menampilkan asal sekolah dari mahasiswa yang memiliki nama sma dan kotanya pemalang	Berhasil
alamat=kalibeber	Menampilkan array kosong	Berhasil
q=SMA Negeri 1 Demak	Menampilkan asal sekolah dari mahasiswa	Berhasil

3.2. Pengujian Kecepatan

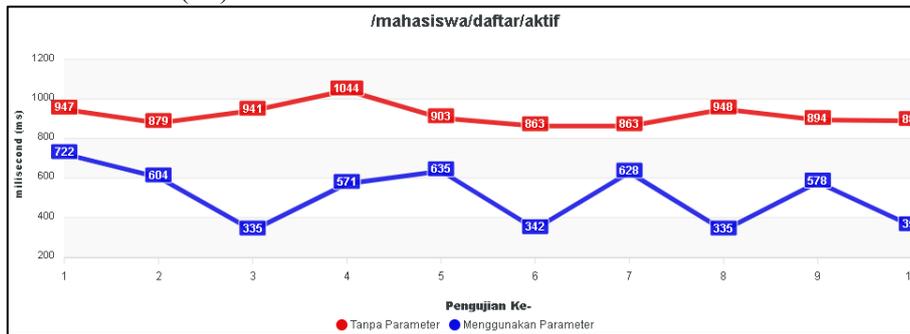
Pengujian berikutnya adalah pengujian kecepatan dari setiap REST API yang telah dibuat sebelumnya, pengujian dilakukan dengan mengambil 3 API Endpoint dari REST API masing-masing server yang dilakukan sebanyak 10 kali baik dengan parameter ataupun tidak, untuk mengetahui rata-rata kecepatan request dalam satuan milisecond.

- 1) Sinta Unissula, pada gambar 3 adalah hasil pengujian API Endpoint “/unissula/penelitian”, yang dilakukan sebanyak 10 kali dengan menggunakan parameter dan tidak menggunakan parameter dengan satuan kecepatan millisecond (ms)



Gambar 3 API Endpoint /unissula/penelitian

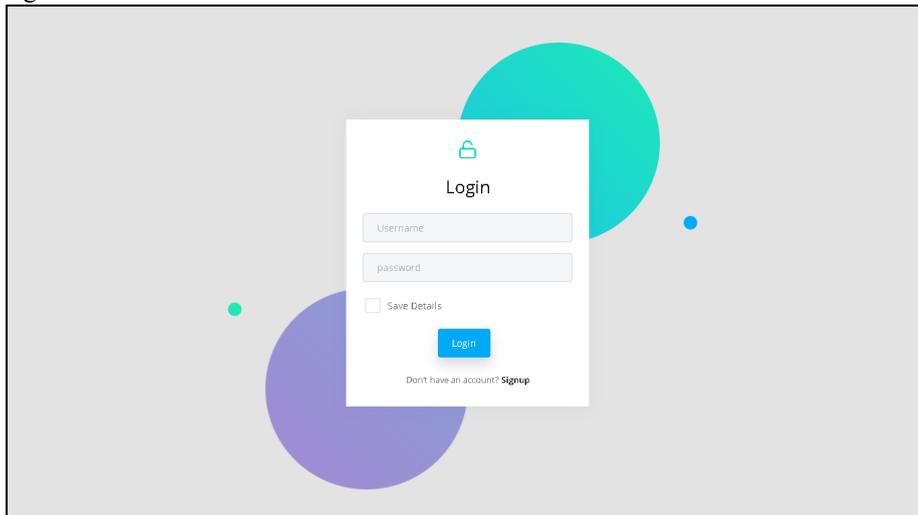
- 2) Unissula, pada gambar 4 adalah hasil pengujian API Endpoint “/mahasiswa/daftar/aktif”, yang dilakukan sebanyak 10 kali dengan menggunakan parameter dan tidak menggunakan parameter dengan satuan kecepatan millisecond (ms)



Gambar 4 API Endpoint /mahasiswa/daftar/aktif

3.3. Implementasi Ke Aplikasi Sistem Monitoring Manajemen YBWSA

Pada tahap ini REST API yang sudah dibuat, diimplementasikan ke aplikasi sistem monitoring manajemen YBWSA. Pada gambar 5 merupakan halaman dari sistem monitoring manajemen dari Yayasan Badan Wakaf Sultan Agung.



Gambar 5 Halaman Login

Pada gambar 6 merupakan halaman dashboard dari sistem monitoring manajemen dari Yayasan Badan Wakaf Sultan Agung, yang didalamnya terdapat data dari Sinta Unissula dan Unissula akademik



Gambar 6 Halaman Dashboard

Pada gambar 7 merupakan halaman daftar dosen dari sistem monitoring manajemen dari Yayasan Badan Wakaf Sultan Agung, data berasal dari API Endpoint /unissula/dosen

ID Sinta	Nama	Fakultas	Sinta Score V2	Sinta Score V2 3 Year
44	IMAM MUCH IBNU SUBROTO	Fakultas Teknologi Industri Teknik Informatika	1232.5	158.5
50	SLAMET IMAM WAHYUDI	Fakultas Teknik Teknik Sipil	1003	443
131	OLIVIA FAHRUNNISA	Fakultas Ekonomi Ilmu Manajemen	805	315.5
1	ANIS MASHDUPOHATUN	Fakultas Hukum Ilmu Hukum	732	433
42	ARIEF MARWANTO	Fakultas Teknologi Industri Teknik Elektro	725.5	220
6200121	AGUNG PUTRA	Fakultas Kedokteran Biomedik	703.5	680.5

Gambar 7 Halaman Daftar Dosen Dari REST API Sinta Unissula

4. KESIMPULAN

Pada penelitian ini telah ditunjukkan bahwa penggunaan teknologi *web services* dengan metode REST API pada prototipe Sistem Monitoring Manajemen Yayasan Badan Wakaf Sultan Agung dapat mengintegrasikan dua sistem lainnya yang berbeda platform dari unit yang berbeda sehingga kebutuhan informasi yang lengkap, tepat dan cepat dapat terpenuhi untuk kebutuhan monitoring.

Pada Pengujian kecepatan, penggunaan parameter dapat membantu mempercepat untuk menampilkan data yang diperlukan, sedangkan tanpa menggunakan parameter server membutuhkan waktu yang lebih lama untuk menampilkan data, pada pengujian pada API Endpoint /unissula/penelitian yang dilakukan sebanyak 10 kali yang masing-masing dalam satuan milisecond (ms). Untuk rata-rata kecepatan tanpa menggunakan parameter adalah 106, 2 milisecond, sedangkan dengan menggunakan parameter “judul=learning&kdProdi=55201” adalah 80 milisecond, Kemudian pada pengujian API Endpoint /mahasiswa/daftar/aktif yang dilakukan sebanyak 10 kali rata-rata kecepatan tanpa menggunakan parameter adalah 917,1 milisecond, sedangkan dengan parameter “idProdi=110604&angkatan=2021” rata-rata kecepatan adalah 510, 7 milisecond.

ACKNOWLEDGEMENTS

Terima kasih kepada Fakultas Teknologi Industri Universitas Islam Sultan Agung atas dukungannya sehingga penelitian ini berjalan dengan lancar. Paper ini merupakan tugas akhir mahasiswa teknik informatika dengan fokus penggunaan *web services* dengan metode REST API untuk memanfaatkan data dari platform yang berbeda, yang digunakan untuk aplikasi sistem monitoring manajemen dari Yayasan Badan Wakaf Sultan Agung.

DAFTAR PUSTAKA

- [1] B. Cooksey, *An Introduction To API'S*. Zapier.Inc, 2014.
- [2] D. H. Setiabudi dan I. K. Julistiono, “Pengujian Penggunaan Simple Object Access Protocol (SOAP) pada Remote Procedure Call (RPC),” *J. Tek. Elektro Univ. Kristen Petra*, vol. 5, no. 1, hal. 38–44, 2005.
- [3] S. Aloysinghe, *RESTful PHP Web Services*. Packt Pub., 2008.
- [4] S. Cirani, G. Ferrari, M. Picone, dan L. Veltri, *Internet of Things Architectures, Protocols and Standards*. Wiley, 2018.
- [5] L. Richardson dan S. Ruby, *RESTful Web Services*. O'Reilly Media, 2008.
- [6] R. Anthus dan D. Prawira, “Rancang Bangun Sistem Informasi Peringatan Dini Kejadian Luar Biasa Menggunakan Model Interoperabilitas Dan Arsitektur Rest,” *J. Komput. dan Apl.*, vol. 08, no. 01, hal. 122–131, 2020.
- [7] W. G. Wardhana, I. Arwani, dan B. Rahayudi, “Implementasi Teknologi Restful Web Service Dalam Pengembangan Sistem Informasi Perekaman Prestasi Mahasiswa Berbasis Website (Studi Kasus : Fakultas Teknologi Pertanian Universitas Brawijaya),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 2, hal. 680–689, 2020.
- [8] M. D. Rafiqi, E. Subyantoro, dan D. K. W., “Implementasi Arsitektur Microservice Pada Aplikasi Online Travel Tourinc,” no. 1, hal. 1–10, 2020.
- [9] F. Al Isfahani, F. Nugraha, R. Mubarak, dan A. Rahmatulloh, “Implementasi Web Service untuk Aplikasi Pemantau Coronavirus Disease 2019,” *Innov. Res. Informatics (Innov.)*, vol. 2, no. 1, hal. 33–39, 2019.
- [10] F. Salamun, P. Studi, T. Informatika, dan U. P. Bandung, “Implementasi Web Service pada Aplikasi Mobile untuk Mendukung Sistem Informasi di Bandung N-Max Community,” hal. 8–9, 2018.