
ANALISIS DAN PENGEMBANGAN API SIAKAD MENGUNAKAN ARSITEKTUR RESTFUL WEB SERVICE PADA INFRASTRUKTUR MICROSERVICE

Dimas Kita Ladiba, Weda Adistianaya Dewa dan Samsul Arifin
STMIK PPKIA Pradnya Paramita (STIMATA)
Jl. Laksda Adi Sucipto 249A, Malang, Jawa Timur
dimaskitaladiba@gmail.com, {weda, samsularifin}@stimata.ac.id

ABSTRAK

Aktivitas pengolahan data akademik di Kampus STIMATA dilakukan melalui banyak aplikasi berbeda dan juga terdapat beberapa pengolahan data yang masih manual. Hal tersebut membuat aktivitas akademik di Kampus STIMATA menjadi kurang efektif. Berdasarkan masalah tersebut, maka perlu pengembangan SIAKAD (Sistem Informasi Akademik) yang terdiri dari dua aplikasi utama berbentuk Web dan Mobile (Android). Kedua aplikasi tersebut berasal dari platform yang berbeda sehingga membutuhkan sebagai sara komunikasinya. API akan menggunakan arsitektur Restful Web Service dan di terapkan di infrastruktur microservice agar API dapat berkomunikasi secara multiplatform dan dapat meminimalisir gangguan teknis. Aplikasi yang dikembangkan dengan metode yang diusulkan dapat memudahkan dalam pengembangan aplikasi dari berbagai platform yang berbeda. Dibuktikan dengan rata-rata besar index interpretasi sebesar 95,3% pada kuesioner yang telah dilakukan. Dan sistem pun sudah bisa digunakan dan hal tersebut dibuktikan dengan tidak adanya error pada pengujian black box

Kata Kunci : SIAKAD, API, Restfull Web Service, Microservice

PENDAHULUAN

Latar Belakang

Berdasarkan wawancara yang dilakukan dengan Kepala dari Divisi Pusat Data di Kampus STIMATA pada tanggal 24 Oktober 2020, untuk saat ini aktivitas pengolahan data akademik dilakukan melalui banyak aplikasi berbeda dan juga terdapat beberapa pengolahan data yang masih manual. Petugas sering melakukan pengolahan data ulang karena beberapa data tidak tersinkron dengan baik. Untuk melakukan pengolahan data akademik hanya bisa dilakukan di lokal komputer Petugas Pusat Data itu saja. Mahasiswa sering kesusahan ketika ingin melihat KRS atau pun KHS karena web yang disediakan terkadang tidak bisa di akses. Pengecekan tagihan atau konfirmasi pembayaran tagihan masih menggunakan chat via WhatsApp yang dirasa kurang efektif dan kurang aman.

SIAKAD (Sistem Informasi Akademik) yang memiliki dua aplikasi utama yang berbentuk *WEB* dan *Mobile* (Android) merupakan salah satu solusinya. Karena SIAKAD memiliki terdiri dari dua platform yang berbeda, maka dari itu dibutuhkan sebuah API sebagai jembatan dalam berkomunikasi. API akan

menggunakan arsitektur *Restful Web Service* sebagai metodenya. API *Web Service* dengan arsitektur *REST* berhasil mencapai tujuan dan sasaran di mana dapat digunakan untuk membuat aplikasi *multiplatform* dengan data yang terintegrasi[1] sehingga arsitektur ini sangat cocok untuk diterapkan pada SIAKAD yang memiliki aplikasi *multiplatform*. Untuk pertukaran datanya akan menggunakan JSON, karena JSON memiliki format penulisan yang lebih sederhana dan didukung oleh banyak *library*[2], maka proses pengunduhan data dari *webservice* lebih cepat. API tersebut akan diterapkan pada infrastruktur *microservice* yang dapat meminimalisir gangguan teknis pada , karena akan dibagi menjadi beberapa modul/layanan.

Tujuan Penelitian

Berdasarkan permasalahan yang telah dirumuskan, maka penelitian ini memiliki tujuan, yaitu : Pengembangan API untuk Sistem Informasi Akademik pada Kampus STIMATA dengan menggunakan arsitektur *Restful Web Service* di infrastruktur *microservice* untuk

memudahkan dalam pengembangan aplikasi dari berbagai platform yang berbeda.

Tinjauan Pustaka

API

API (*Application Programming Interface*) atau biasa disebut merupakan sebuah sistem yang menjembatani komunikasi antara satu aplikasi dengan aplikasi lainnya[3]. Di dalam sebuah biasanya mencakup spesifikasi untuk fungsi / modul, struktur data, kelas objek, dan variabel.

Dalam pengiriman datanya, API memiliki banyak format yang bisa digunakan, namun format yang paling optimal adalah JSON. Karena JSON memiliki format penulisan yang lebih sederhana dan didukung oleh banyak *library* [2],

API juga membutuhkan *endpoint* sebagai URL akses dari tersebut, *endpoint* dan sistem *proxy* dijelaskan, di mana *user* dapat mengelola dan mengaktifkan lalu menyebabkan eksekusi kode program pada sistem pihak ketiga atau jarak jauh[4]. Sehingga aplikasi akan mengakses sebuah URL untuk berkomunikasi dengan aplikasi lainnya melalui sebuah API. Dilihat dari uraian di atas API sangat diperlukan ketika akan membangun sebuah sistem informasi *multi-platform*.

SIAKAD

Sistem informasi akademik (SIKAD) merupakan layanan akademik yang diperuntukkan bagi mahasiswa dalam mengakses informasi yang berkaitan dengan catatan akademik selama proses perkuliahan[6]. Pengelolaan bidang Akademik akan menjadi lebih efektif dan efisien setelah dibantu dengan penggunaan teknologi informasi dalam bentuk SIKAD[7]. SIKAD pada Perguruan Tinggi akan mengelola kegiatan akademik seperti KHS, KRS, administrasi dan kegiatan akademik lainnya. SIKAD bertujuan untuk mendukung penyelenggaraan pendidikan, sehingga dalam proses pendidikan dapat disediakan layanan informasi yang lebih baik kepada mahasiswanya[8]. SIKAD secara

simultan dan parsial berpengaruh signifikan berpengaruh pada kepuasan siswa[6].

RESTFUL WEB SERVICE

Restful Web Service merupakan salah satu jenis *Web Service*, sedangkan *Web Service* adalah aplikasi client server yang menyediakan media untuk berkomunikasi antara aplikasi yang berbeda jaringan [9]. Terdapat dua arsitektur API yang populer, yaitu *SOAP* dan *Restful.SOAP* pertama kali terlihat pada Juni 1998 dan rilis stabil ke publik pada 27 April 2007[10], sedangkan *Restful Web Service* diperkenalkan pada tahun 2000 oleh Roy Fielding pada acara disertasi doktor miliknya[11]. *Restful Web Service* memiliki beberapa metode HTTP dalam penggunaannya :

- GET digunakan untuk mengambil data dari database atau resource.
- POST digunakan untuk menambahkan data atau memverifikasi data ke dalam database atau resource.
- PUT digunakan untuk mengganti data di dalam database atau resource.
- DELETE digunakan untuk menghapus data di dalam database atau resource.

Dalam sisi teknologi, proses, kemudahan penggunaan dan desainnya, *Restful* lebih unggul daripada *SOAP* dan *Restful* diprediksi akan terus mendominasi lebih banyak ruang teknologi di tahun-tahun mendatang dalam hal membangun sebuah backend *Restful*[12].

MICROSERVICE

Sebelum adanya arsitektur *microservice*, terdapat arsitektur *monolithic*. *Monolithic* adalah sebuah sistem yang semua komponennya menjadi satu kesatuan. Ketika aplikasi di akses oleh banyak *user*, maka performa akan menurun. Lalu di saat akan ada perubahan teknologi atau skema bisnis aplikasi akan membuat pengembang kesulitan. Di dalam satu arsitektur *monolithic* bisa terdapat puluhan atau ratusan *service* yang dapat membuat pekerjaan tim pengembang kesulitan, karena itu banyak perusahaan yang bergerak untuk mengubah jenis arsitekturnya ke arsitektur *microservice* untuk memungkinkan tim

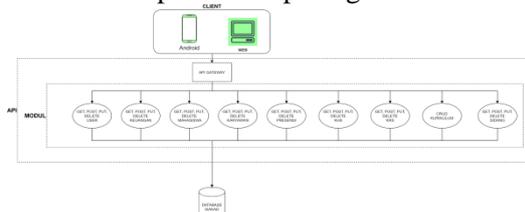
pengembang mereka dapat berkoordinasi dengan mudah satu sama lain[14].

Jadi arsitektur *microservice* menggunakan desain aplikasi yang memecah-mecah aplikasi berdasarkan kegunaan atau fungsinya atau modulnya. Jika *microservice* diterapkan pada API, maka API akan dipecah sesuai modul tersebut. Karena API akan dipecah-pecah maka API tersebut dapat dibuat menggunakan bahasa pemrograman yang berbeda-beda pada setiap modul[15].

METODE PENELITIAN

Restful Web Service

Solusi yang diusulkan untuk memecahkan masalah yang telah dideskripsikan adalah, perlu dibuatnya API yang akan menjembatani komunikasi antar aplikasi dengan *database* pada server. API tersebut akan menggunakan arsitektur *Restful Web Service* yang akan diimplementasikan pada infrastruktur *microservice*. Model alur kerja sistem yang diusulkan dapat dilihat pada gambar 1.

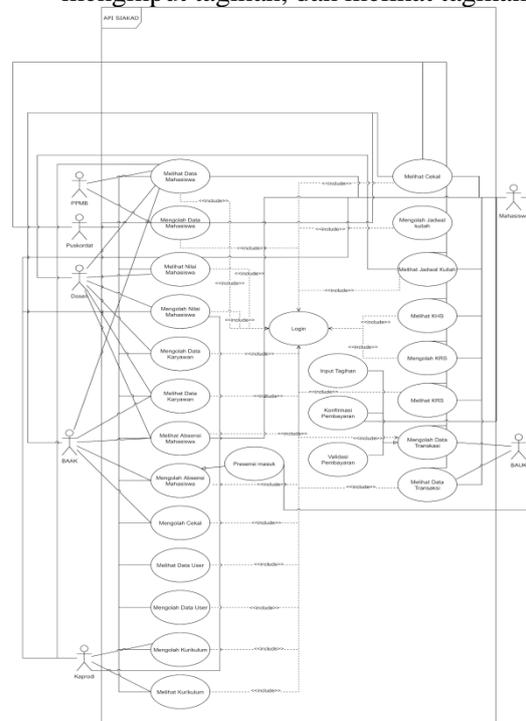


Gambar 1. Alur API SIAKAD

Use Case Diagram

1. Puskordat merupakan aktor yang bisa menggunakan semua fungsi dari SIAKAD. Dikarenakan Puskordat merupakan bagian yang bertanggung jawab atas segala kegiatan akademik pada kampus STIMATA.
2. PPMB merupakan panitia yang ditunjuk oleh pihak kampus untuk melakukan pencatatan mahasiswa baru. PPMB di sini bisa mengakses dan mengolah data mahasiswa.
3. Dosen bisa melihat data mahasiswa, melihat nilai mahasiswa yang terdaftar dalam mata kuliahnya, melihat data pribadi, melihat absensi yang terdaftar dalam mata kuliahnya, dan mengolah nilai mahasiswa yang terdaftar dalam mata kuliahnya.

4. BAAK dapat menggunakan fungsi mengolah cekal, mengolah absensi mahasiswa, mengolah jadwal kuliah, melihat data mahasiswa, melihat cekal, melihat absensi mahasiswa, dan melihat jadwal kuliah
5. Kaprodi dapat melihat data mahasiswa, melihat nilai mahasiswa, melihat kurikulum, mengolah nilai mahasiswa, dan mengolah kurikulum.
6. Mahasiswa dapat melihat data dirinya, melihat nilai, melihat jadwal, melihat absen, presensi absen, melihat tagihan, melakukan konfirmasi pembayaran dan mengolah data dirinya.
7. BAUK dapat mengolah data tagihan, melakukan validasi pembayaran, menginput tagihan, dan melihat tagihan.



Gambar 2. Use Case Diagram

HASIL DAN PEMBAHASAN

Hasil Pengujian Sistem

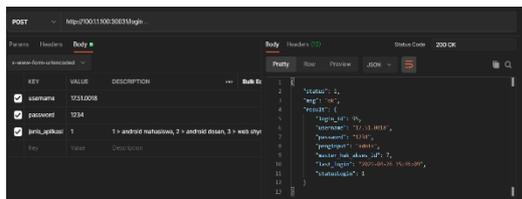
Pengujian akan dilakukan menggunakan metode *black box testing* ke seluruh API.

Model User

Tabel 1. Black Box Modul User

End Point	Action	Hasil
user/login/	Melakukan login menggunakan NIM 17.51.0018	Sukses
user/register/	Menampilkan semua data user	Sukses
user/register/	Menambahkan data test ke register	Sukses
user/register/	Mengganti data test menjadi testing	Sukses
user/register/	Menghapus data testing	Sukses
user/role/	Mengambil semua data role	Sukses
user/role/	Menambah data test ke role	Sukses
user/role/	Mengedit data test menjadi testing ke role	Sukses
user/role/	Menghapus data testing	Sukses

Hasil dari testing *black box* pada modul ini secara keseluruhan sukses. Sebagai contoh *request* login pada baris nomor 1, peneliti menggunakan akun mahasiswa dengan NIM 17.51.0018 untuk melakukan login. Data yang dimasukkan adalah *username*, *password* dan jenis aplikasi. Jika, *user* terverifikasi maka response dari seperti pada gambar 8 dibawah.



Gambar 3. Tampilan Postman Request Login

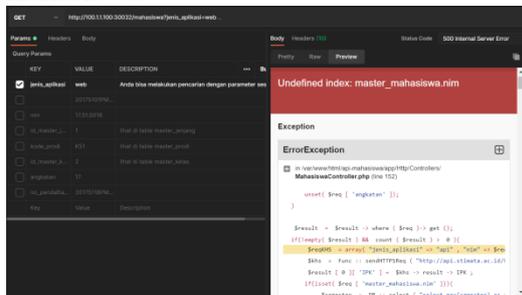
Modul Mahasiswa

Tabel 2. Black Box Modul Mahasiswa

End Point	Action	Hasil
mahasiswa/maba	Menampilkan semua data maba, dengan mengosongkan parameter <i>request</i>	Sukses
mahasiswa/maba	Menambahkan data maba test	Sukses
mahasiswa/maba	Nama ibu maba test diubah menjadi Lala	Sukses
mahasiswa/maba	Maba test dihapus	Sukses
mahasiswa	Menampilkan semua data mahasiswa	Gagal
mahasiswa	Menampilkan semua data mahasiswa	Sukses
mahasiswa	Menambahkan data mahasiswa dengan no pendaftaran 20163101PMBMB01	Sukses
mahasiswa	Mengedit tahun masuk nim 27510019	Sukses
mahasiswa	Menghapus mahasiswa dengan nim 27510019	Sukses
mahasiswa/master/status daftar	Menampilkan semua data	Sukses
mahasiswa/master/status daftar	Menambah data test	Sukses
mahasiswa/master/status daftar	Mengganti data nama test menjadi testing	Sukses
mahasiswa/master/status daftar	Menghapus data testing	Sukses
mahasiswa/master/status	Menampilkan semua data	Sukses

End Point	Action	Hasil
mahasiswa/ master/status	Menambah data test	Sukses
mahasiswa/ master/status	Mengganti data nama test menjadi testing	Sukses
mahasiswa/ master/status	Menghapus data testing	Sukses
mahasiswa/s emester	Menampilkan semester terakhir dari nim 17.51.0018	Sukses

Terdapat satu *request* yang mengalami kegagalan, yaitu *requestget all mahasiswa*.



Gambar 4. Hasil Postman Request Get All Mahasiswa

Error terjadi karena variabel *master_mahasiswa.nim* tidak ditemukan pada variabel *\$req*. Hal ini disebabkan karena tidak adanya variabel *nim* saat melakukan *request* ke . Solusinya adalah dengan cara melakukan pengecekan terlebih dahulu, apakah pada saat *request* mengirimkan *nim* atau tidak.

Modul Karyawan

Tabel 3. Black Box pada Modul Karyawan

End Point	Action	Hasil
dosen/	Menampilkan semua data	Sukses
dosen/	Menambah data test	Sukses

End Point	Action	Hasil
dosen/	Mengganti data nama test menjadi testing	Sukses
dosen/	Menghapus data testing	Sukses
karyawan/	Menampilkan semua data	Sukses
karyawan/	Menambah data test	Sukses
karyawan/	Mengganti data nama test menjadi testing	Sukses
karyawan/	Menghapus data testing	Sukses
karyawan/d osen/status	Menampilkan semua data	Sukses
karyawan/d osen/status	Menambah data test	Sukses
karyawan/d osen/status	Mengganti data nama test menjadi testing	Sukses
karyawan/d osen/status	Menghapus data testing	Sukses
karyawan/ja batan/	Menampilkan semua data	Sukses
karyawan/ja batan/	Menambah data test	Sukses
karyawan/ja batan/	Mengganti data nama test menjadi testing	Sukses
karyawan/ja batan/	Menghapus data testing	Sukses

End Point	Action	Hasil
karyawan/dosen/pangkat	Menampilkan semua data	Sukses
karyawan/dosen/pangkat	Menambah data test	Sukses
karyawan/dosen/pangkat	Mengganti data nama test menjadi testing	Sukses
karyawan/dosen/pangkat	Menghapus data testing	Sukses
karyawan/dosen/mahasiswa	Menampilkan anak didik dari dosen seng nip 14.02.25.002	Sukses

Hasil dari pengujian *black box* pada modul karyawan menunjukkan bahwa *request* dapat dijalankan tanpa ada kendala.

Modul Keuangan

Tabel 4. *Black Box* pada Modul Keuangan

End Point	Action	Hasil
keuangan/tagihan/	Menampilkan semua tagihan	Sukses
keuangan/tagihan/	Menambah tagihan dengan NIM 17.51.0018, sejumlah 270000	Sukses
keuangan/tagihan/	Mengganti qty tagihan 17.51.0018 menjadi 1 dari 2	Sukses

End Point	Action	Hasil
keuangan/tagihan/	Menghapus salah satu tagihan dari nim 17.51.0018	Sukses
keuangan/pembayaran/	Menampilkan pembayaran dengan NIM 17.51.0018	Sukses
keuangan/pembayaran/	Melakukan pembayaran sejumlah 135000 pada NIM 17.51.0018	Sukses
keuangan/pembayaran/update	Update gambar pembayaran NIM 17.51.0018	Sukses
keuangan/pembayaran/	Delete salah satu pembayaran NIM 17.51.0018	Sukses
keuangan/master/biaya	Menampilkan semua data	Sukses
keuangan/master/biaya	Menambahkan biaya test	Sukses
keuangan/master/biaya	Mengedit data test menjadi testing	Sukses
keuangan/master/biaya	Menghapus data testing	Sukses
keuangan/master/jenis pembayaran	Menampilkan semua data	Sukses

End Point	Action	Hasil
keuangan/ master/jeni spembayar an	Menambah data test pada jenis pembayaran	Sukses
keuangan/ master/jeni spembayar an	Mengedit data test menjadi testing	Sukses
keuangan/ master/jeni spembayar an	Menghapus data testing	Sukses
keuangan/t agihan/nim	Menampilkan tagihan dari NIM 17.51.0018	Sukses
keuangan/ pembayara n/bukti	Menampilkan bukti bayar dari NIM 17.51.0018	Sukses

Tabel di atas menunjukkan bahwa dari Modul Pembayaran dapat menghasilkan data *response* yang sesuai dengan *request* yang diberikan.

Modul Presensi

Tabel 5. Hasil Postman pada Modul Presensi

End Point	Action	Hasil
presensi/m ahasiswa/a bsen	Menampilkan data presensi mahasiswa 17.51.0018	Sukses
presensi/m ahasiswa/a bsen	Melakukan absen dengan token 6f19cecd9668dd57 201620cf65333fd3	Sukses

End Point	Action	Hasil
presensi/m ahasiswa/a bsen/manu al	Absensi manual ke NIM 17.51.0018, dengan status hadir	Sukses
presensi/ge ttoken/ token/	Berhasil mendapatkan token absensi	Sukses
presensi/m aster/wakt upresensi	Mengambil semua data	Sukses
presensi/ab sen	Menampilkan laporan absensi sesuai minggu dan krs	Sukses
presensi/la poran/abse n	Menampilkan Laporan absensi mahasiswa 17.51.0018 dan semester 8	Sukses
presensi/la poran/abse n/krs	Menampilkan laporan absensi sesuai krs id	Sukses

Tabel di atas menunjukkan bahwa dari Modul Presensi dapat menghasilkan data *response* yang sesuai dengan *request* yang diberikan.

Modul KRS

Tabel 6. Black Box pada Modul KRS

End Point	Action	Hasil
krs/ krs/	Menampilkan KHS pada tahun akademik 2020- 2021	Sukses

End Point	Action	Hasil
krs/	error	Gagal
krs/	Mengganti ketua kelas pada krs Perilaku Dalam Berorganisasi	Sukses
krs/mahasiswa/	Menampilkan KRS dengan NIM 17.51.0018 dan semester 8	Sukses
krs/mahasiswa/	Menambahkan KRS ke NIM 17.51.0018 dan semester 8	Sukses
krs/detail	Menampilkan KRS detail dengan tahun akadmeik 2020-2021	Sukses
krs/peserta	Menampilkan KRS detail dari KRS Perilaku Dalam Berorganisasi	Sukses
krs/terbaru	Menampilkan KRS terbaru	Sukses
krs/mahasiswa/totaltempuh	Menampilkan krs yang ditempuh oleh NIM 17.51.0018	Sukses

End Point	Action	Hasil
krs/mahasiswa/semester	Menampilkan krs pada setiap semester oleh NIM 17.51.0018	Sukses
krs/dosen	Menampilkan KRS yang ditempuh oleh dosen	Sukses
krs/dosen/otorisasi	Melakukan otorisasi KRS dengan NIM 17.51.0013	Sukses
jadwal/mahasiswa	Mendapatkan jadwal matkul dengan NIM 17.51.0018 dan hari 2021-04-21	Sukses
jadwal/dosen	Mendapatkan jadwal matkul dengan NIP 00.12.01.004 dan hari 2021-03-17	Sukses

Data diatas menunjukkan bahwa terdapat satu kendala pada krs dengan *methodPOST*. tersebut berfungsi untuk menambah krs yang bisa diambil oleh mahasiswa.

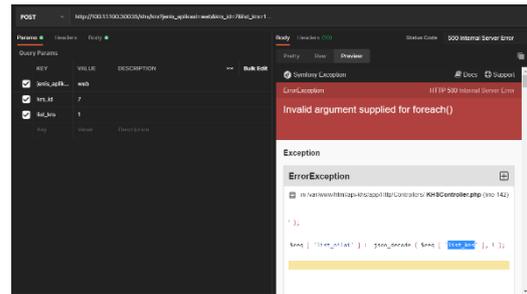
Modul KHS

Tabel 7. Black Box pada Modul KHS

End Point	Action	Hasil
khs/master/tipenilai	Menampilkan semua data	Sukses

End Point	Action	Hasil
khs/krs	Menampilkan data dengan matkul Perilaku dalam Berorganisasi dengan tahun akademik 2020-2021 dan tugas 1	Sukses
khs/krs		Gagal
nilai/kaprodi	Menambah nilai mata kuliah Perilaku Dalam Berorganisasi pada NIM 17.51.0018 dengan nilai B	Sukses
khs/salinan nilai	Menampilkan salinan nilai NIM 17.51.0018	Sukses
khs/	Menampilkan KHS dengan NIM 17.51.00018 pada semester 7	Sukses
ipk	Menampilkan IPK dari NIM 17.51.0018	Sukses

Data diatas menunjukkan bahwa terdapat satu kendala pada khs/krs dengan *methodPOST*. tersebut berfungsi untuk menambahkan nilai matakuliah mahasiswa oleh dosen. Gambar 5 menunjukkan kesalahan yang terjadi pada .



Gambar 5. Tampilan Postman pada Request API Tambah KRS

Berdasarkan gambar di atas, kesalahan terjadi karena *Invalid argument supplied for foreach()* yang disebabkan karena kesalahan penulisan variabel `list_krs` yang seharusnya `list_nilai`.

Modul Sidang

Tabel 8. Black Box pada Modul Sidang

End Point	Action	Hasil
master/tipe nilai	Mengambil semua data tipe nilai	Sukses
master/tipe nilai	Menambah tipenilai test	Sukses
master/tipe nilai	mengubah tipenilai test menjadi testing	Sukses
master/tipe nilai	menghapus tipenilai testing	Sukses
master/tug asakhir	Mengambil semua data tugas akhir	Sukses
master/tug asakhir	Menambah tugas akhir test	Sukses
master/tug asakhir	mengubah tugas akhir test menjadi testing	Sukses
master/tug asakhir	menghapus tugas akhir testing	Sukses

End Point	Action	Hasil
nilai/tugas akhir	Mengambil semua data master tugas akhir	Sukses
nilai/tugas akhir	Menambah master tugas akhir test	Sukses
nilai/tugas akhir	mengubah master tugas akhir test menjadi testing	Sukses
nilai/tugas akhir	menghapus master tugas akhir testing	Sukses
tugasakhir/mahasiswa	Mengambil semua data tugas akhir mahasiswa	Sukses
tugasakhir/mahasiswa	Menambah tugas akhir mahasiswa test	Sukses
tugasakhir/mahasiswa	mengubah tugas akhir mahasiswa test menjadi testing	Sukses
tugasakhir/mahasiswa	menghapus tugas akhir mahasiswa testing	Sukses

Tabel di atas menunjukkan bahwa dari Modul Sidang dapat menghasilkan data *response* yang sesuai dengan *request* yang diberikan.

Pengujian Akurasi

Pengujian dilakukan dengan melakukan kuesioner dan dilanjutkan dengan perhitungan menggunakan Skala Likert. Terdapat 2 responden khusus yang

sudah mengisi kuesioner penelitian ini, hasil kuesioner dapat dilihat pada tabel 24

Tabel 9. Hasil Kuesioner

Rsp	App	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
A	Web Siakad Internal	5	5	5	5	5	5	5	5	5	4	5	5	5	5	5
B	Android Siakad	5	5	4	5	5	5	4	5	5	5	5	5	1	5	5

Data pada tabel 24 menunjukkan bahwa API dapat memudahkan responden dalam melakukan pengembangan aplikasi dari platform yang berbeda dengan rata-rata indeks interpretasi sebesar 95.3%.

PENUTUP

Berdasarkan analisis hasil pengujian pada eksperimen yang telah dilakukan, maka dapat diambil kesimpulan bahwa penerapan *Restfull Web Service* pada infrastruktur *microservice*, dapat memudahkan dalam pengembangan aplikasi dari berbagai platform yang berbeda. Dibuktikan dengan rata-rata besar index interpretasi sebesar 95,3% pada kuesioner yang telah dilakukan. Dan sistem pun sudah bisa digunakan dan walaupun masih terdapat *error* pada bagian tertentu.

DAFTAR PUSTAKA

- [1] R. Choirudin dan A. Adil, "Implementasi Rest Api Web Service dalam Membangun Aplikasi Multiplatform untuk Usaha Jasa," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 18, no. 2, 2019.
- [2] G. Barbaglia, S. Murzilli dan S. Cudini, "Definition of REST web services with JSON schema," *Software - Practice and Experience*, vol. 47, no. 6, 2017.
- [3] P. Mahiddini, "DYNAMIC APPLICATION PROGRAMMING INTERFACE PUBLICATION FOR PROVIDING WEB SERVICES," *United States Patent*, 2017.

-
- [4] J. P. Thompson, "Endpoint management system providing an application programming interface proxy service," 3 2018.
- [5] Y. Li, N. R. Katsipoulakis, B. Chandramouli, J. Goldstein dan D. Kossmann, "Mison," Proceedings of the VLDB Endowment, vol. 10, no. 10, pp. 1118-1129, 2017.
- [6] W. O. T. Hayati, A. Palilati dan E. Sukotjo, "PENGARUH SISTEM INFORMASI AKADEMIK (SIKAD) DAN KUALITAS PELAYANAN ADMINISTRASI TERHADAP KEPUASAN MAHASISWA," Jurnal Manajemen, Bisnis dan Organisasi (JUMBO), vol. 2, no. 1, 2018.
- [7] Marisa dan Harleni, "SISTEM INFORMASI AKADEMIK (SIKAD) STIKES PERINTIS PADANG," Jurnal Teknologi Informasi dan Ilmu Komputer, vol. 5, no. 3, 2018.
- [8] M. B. Suryawan dan P. Prihandoko, "Evaluasi Penerapan SIKAD Politeknik Negeri Madiun Menggunakan Pendekatan TAM dan EUCS," Creative Information Technology Journal, vol. 4, no. 3, p. 233, 2018.
- [9] A. Soni dan V. Ranga, "API features individualizing of web services: REST and SOAP," International Journal of Innovative Technology and Exploring Engineering, vol. 8, no. 9 Special Issue, 2019.
- [10] "SOAP," Wikipedia, [Online]. Available: <https://en.wikipedia.org/wiki/Soap>.
- [11] X. Chen, Z. Ji, Y. Fan dan Y. Zhan, "Restful API Architecture Based on Laravel Framework," Journal of Physics: Conference Series, vol. 910, p. 012016, 2017.
- [12] F. Halili dan E. Ramadani, "Web Services: A Comparison of Soap and Rest Services," Modern Applied Science, vol. 12, no. 3, p. 175, 2018.
- [13] J. Muittari, "Modern web back-end," 2020. [Online].
- [14] O. Al-Debagy dan P. Martinek, "A Comparative Review of Microservices and Monolithic Architectures," 2018. [Online].
- [15] B. Terzić, V. Dimitrieski, S. Kordić, G. Milosavljević dan I. Luković, "Development and evaluation of MicroBuilder: a Model-Driven tool for the specification of REST Microservice Software Architectures," Enterprise Information Systems, vol. 12, no. 8-9, pp. 1034-1057, 2018.
- [16] R. A. Setyawan, "Pengukuran Usability Website E-Commerce Sambal Nyoss Menggunakan Metode Skala Likert," Compiler, vol. 7, no. 1, 2018.