

# Penerapan Web Service API dalam Sistem Informasi Akademik (SIKAD) pada IAIN Takengon

M. Tajul Munandar<sup>1</sup>, Salahuddin<sup>2\*</sup>, Mahdi<sup>3</sup>

<sup>1,2,3</sup> *Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe  
Jln. B.Aceh Medan Km.280 Buket Rata 24301 INDONESIA*

<sup>1</sup>tajulmunandar701@gmail.com

<sup>2\*</sup> salahuddintik@pnl.ac.id (penulis korespondensi)

<sup>3</sup>mahdi@pnl.ac.id

**Abstrak**—Sistem Informasi Akademik (SIKAD) merupakan salah satu solusi teknologi penting dalam menunjang operasional akademik di perguruan tinggi. Penelitian ini bertujuan untuk mengembangkan dan menguji integrasi SIKAD dengan aplikasi E-Modul dan Simulasi-SIKAD melalui implementasi Web Service API. Integrasi ini dilakukan untuk mengurangi redundansi data dan meningkatkan efisiensi sinkronisasi data akademik antara SIKAD dan Forlap PDDIKTI. Metode yang digunakan dalam penelitian ini melibatkan perancangan arsitektur sistem berbasis API yang menghubungkan SIKAD dengan berbagai aplikasi, serta pengujian white box untuk memastikan kehandalan dan kebenaran jalur data. Hasil implementasi menunjukkan bahwa integrasi ini tidak hanya berhasil mengurangi redundansi data di E-Modul, tetapi juga mendukung sinkronisasi data yang lebih efisien antara SIKAD dan Forlap PDDIKTI. Selain itu, desain arsitektur yang baik memungkinkan sistem menangani berbagai kebutuhan akademik secara terstruktur, mendukung aktivitas operasional perguruan tinggi, serta memfasilitasi pertukaran informasi yang optimal antara sistem internal kampus dan pihak eksternal. Hal ini membuktikan bahwa integrasi berbasis Web Service API mampu meningkatkan kehandalan, ketersediaan, dan efisiensi pengelolaan informasi akademik di perguruan tinggi.

**Kata Kunci**—SIKAD, API, integrasi sistem, E-Modul, Forlap PDDIKTI, sinkronisasi data

**Abstract**—The Academic Information System (SIKAD) is an essential technological solution for supporting academic operations in higher education institutions. This research aims to develop and test the integration of SIKAD with the E-Module and SIKAD Simulation applications through the implementation of a Web Service API. The integration is designed to reduce data redundancy and enhance the efficiency of academic data synchronization between SIKAD and the Forlap PDDIKTI system. The methods used in this research involve designing an API-based system architecture that connects SIKAD with various applications, as well as conducting white-box testing to ensure the reliability and correctness of data pathways. The implementation results demonstrate that this integration not only successfully reduces data redundancy in the E-Module but also supports more efficient data synchronization between SIKAD and Forlap PDDIKTI. Additionally, the well-structured system architecture enables the system to handle various academic needs more effectively, supports the operational activities of higher education institutions, and facilitates optimal information exchange between internal campus systems and external parties. This proves that Web Service API-based integration can improve the reliability, availability, and efficiency of academic information management in higher education institutions.

**Keywords**—SIKAD, API, system integration, E-Module, Forlap PDDIKTI, data synchronization

## I. PENDAHULUAN

Perkembangan teknologi informasi telah memberikan dampak yang signifikan dalam berbagai sektor, termasuk pendidikan. Perguruan tinggi, sebagai salah satu lembaga yang kompleks, memerlukan sistem informasi yang mampu mengelola data akademik secara efisien dan terintegrasi. Dalam konteks ini, Sistem Informasi Akademik (SIKAD) hadir sebagai solusi yang diterapkan untuk mengatasi tantangan pengelolaan data akademik, termasuk pengelolaan jadwal perkuliahan, pendaftaran mata kuliah, hingga pelaporan nilai mahasiswa. Implementasi SIKAD bertujuan untuk meningkatkan efisiensi operasional, aksesibilitas informasi yang lebih cepat, serta akurasi data bagi mahasiswa, dosen, dan staf administrasi [1].

SIKAD umumnya memanfaatkan teknologi komputer baik perangkat keras (*hardware*) maupun perangkat lunak (*software*), yang mengubah proses manual menjadi otomatis, sehingga seluruh proses akademik dapat dikelola dengan lebih

efektif. Meskipun penerapan SIKAD telah banyak diadopsi, beberapa institusi seperti Institut Agama Islam Negeri (IAIN) Takengon masih menghadapi tantangan dalam hal integrasi dan efisiensi sistem yang belum sepenuhnya optimal. Masalah yang umum terjadi antara lain adalah ketidakakuratan data, keterlambatan dalam proses pelaporan, serta kurangnya integrasi antara SIKAD dan sistem lain yang digunakan oleh perguruan tinggi [2].

Untuk mengatasi masalah tersebut, penggunaan *Web Service API* menjadi solusi yang penting dalam integrasi sistem. *Web Service API* memungkinkan SIKAD berkomunikasi dengan berbagai aplikasi eksternal, seperti sistem keuangan, perpustakaan, atau E-Modul, dengan memanfaatkan protokol standar seperti *REST* atau *SOAP*. Penggunaan *API* ini memastikan pertukaran data yang *real-time* dan akurat antar sistem, yang pada akhirnya mempercepat proses pengambilan keputusan dan

meningkatkan keandalan data yang diterima oleh mahasiswa, dosen, maupun staf administrasi [3].

Di perguruan tinggi seperti IAIN Takengon, penggunaan *Web Service API* akan memudahkan integrasi SIAKAD dengan aplikasi lain, serta mengurangi potensi kesalahan yang terjadi pada proses manual. Sebagai contoh, *API* memungkinkan sinkronisasi data akademik dengan aplikasi E-Modul, sehingga informasi terkait pendaftaran mata kuliah, kehadiran, dan nilai dapat diperbarui secara otomatis tanpa intervensi manual [4].

Selain meningkatkan efisiensi operasional, integrasi *API* juga dapat meminimalisir redundansi data dan inkonsistensi informasi, karena semua aplikasi mengakses data dari satu sumber yang sama. Dengan adanya sistem yang terintegrasi, pengelolaan data akademik menjadi lebih transparan, terorganisir, dan akuntabel. Hal ini akan memudahkan institusi dalam memonitor dan mengelola kinerja akademik, serta memberikan layanan yang lebih baik kepada mahasiswa dan dosen [5].

Dengan demikian, implementasi *Web Service API* diharapkan dapat mendukung SIAKAD IAIN Takengon untuk menjadi sistem yang lebih efisien dan responsif, serta mendorong terciptanya ekosistem pendidikan tinggi yang lebih terintegrasi dan modern.

## II. METODOLOGI PENELITIAN

### A. *Web Service*

*Web Service* adalah sistem perangkat lunak yang didesain untuk mendukung interaksi antar mesin melalui jaringan komputer. Dengan kata lain, semua tipe perangkat lunak yang berbeda dapat saling terhubung dan bertukar informasi serta data yang dibutuhkan melalui protokol dan standar tertentu. *Web Service* memanfaatkan teknologi seperti *HTTP* dan *XML* untuk mengirimkan data antar sistem secara platform-independent. Hal ini memungkinkan sistem yang berjalan di berbagai platform (Windows, Linux, iOS) untuk tetap dapat saling berkomunikasi tanpa tergantung pada bahasa pemrograman atau arsitektur sistem yang digunakan.

Terdapat dua arsitektur utama yang sering digunakan dalam pengembangan *Web Service*, yaitu *SOAP (Simple Object Access Protocol)* dan *REST (Representational State Transfer)*. *SOAP* adalah protokol berbasis *XML* yang digunakan untuk melakukan pertukaran informasi antar sistem. *SOAP* lebih sering digunakan dalam sistem yang membutuhkan keamanan tingkat tinggi, karena mendukung *WS-Security*, yang menyediakan fitur seperti integritas pesan dan otentikasi pesan [6].

Di sisi lain, *REST* adalah arsitektur yang lebih ringan dan sering digunakan dalam pengembangan aplikasi berbasis web modern. *REST* memanfaatkan metode *HTTP* standar (seperti *GET*, *POST*, *PUT*, dan *DELETE*) untuk melakukan operasi pada sumber daya, dan umumnya data dikirimkan dalam format *JSON* atau *XML*. *REST* lebih sederhana dibandingkan *SOAP*, dan lebih cocok untuk aplikasi yang memerlukan respons cepat dan ringan, seperti aplikasi mobile dan web [7].

Keuntungan *RESTful Web Service*

- 1) Sederhana dan Ringan: *REST* memanfaatkan *HTTP* sebagai protokol dasar, sehingga lebih mudah diimplementasikan pada perangkat apa pun yang mendukung *HTTP*.
- 2) Pemrosesan Data Real-Time: *REST* memungkinkan integrasi data secara real-time, yang merupakan salah satu alasan utama mengapa *REST* sering digunakan dalam aplikasi yang membutuhkan pembaruan data secara terus-menerus, seperti SIAKAD.
- 3) Scalability (Skalabilitas): *REST* dirancang untuk mendukung skala besar dan memungkinkan sistem untuk dikembangkan atau diperluas tanpa memerlukan perubahan besar pada arsitektur dasarnya.
- 4) Fleksibilitas Format Data: *REST* memungkinkan data dikirim dalam berbagai format, meskipun yang paling umum digunakan adalah *JSON* karena kesederhanaan dan efisiensinya [8].

*RESTful Web Service* sangat cocok diterapkan dalam Sistem Informasi Akademik (SIAKAD) untuk mengelola data akademik yang membutuhkan kecepatan akses dan efisiensi komunikasi antar sistem yang berbeda. Dengan *Web Service*, integrasi antara berbagai aplikasi di lingkungan perguruan tinggi, seperti sistem pembayaran, perpustakaan, dan modul e-learning, dapat dilakukan dengan mudah [9]. Hal ini memungkinkan peningkatan akurasi dan kecepatan pengambilan keputusan di perguruan tinggi [10].

### B. Teknik Pengumpulan Data

Pengumpulan data yang dilakukan dengan menggunakan beberapa Teknik berikut.

- 1) Observasi langsung di lokasi IAIN Takengon dilakukan untuk memahami proses dan alur kerja sistem akademik secara real-time. Observasi ini memberikan wawasan mengenai bagaimana sistem informasi akademik yang sudah ada beroperasi, kelemahan yang dialami, serta kebutuhan spesifik yang belum terpenuhi. Dengan pendekatan ini, peneliti dapat mengidentifikasi hambatan teknis dan operasional yang ada dalam penggunaan sistem saat ini, yang nantinya dapat menjadi dasar untuk pengembangan solusi berbasis *Web Service*. Observasi juga memberikan gambaran faktual mengenai kondisi infrastruktur teknologi informasi yang mendukung implementasi sistem tersebut.
- 2) Wawancara mendalam dilakukan dengan para pemangku kepentingan utama, seperti dosen, mahasiswa, staf akademik, dan administrasi di IAIN Takengon. Melalui wawancara ini, diperoleh pandangan subjektif dari berbagai perspektif terkait efektivitas sistem informasi akademik yang sedang berjalan. Wawancara bertujuan untuk menggali kebutuhan, harapan, dan masalah yang dihadapi oleh pengguna dalam aktivitas akademik harian mereka. Metode ini membantu untuk memastikan bahwa solusi yang dikembangkan tidak hanya berdasarkan aspek teknis, tetapi juga mempertimbangkan kebutuhan pengguna secara holistik.

- 3) Studi pustaka dilakukan dengan menelaah berbagai literatur terkait sistem informasi akademik, web services, dan integrasi sistem dalam konteks pendidikan tinggi. Peneliti mencari rujukan dari jurnal, buku, dan sumber terpercaya lainnya untuk memastikan bahwa solusi yang dirancang didasarkan pada *best practices* yang telah terbukti efektif di berbagai institusi pendidikan. Literatur yang digunakan mencakup teori-teori tentang arsitektur sistem informasi, keamanan data, dan interoperabilitas antar sistem untuk mendukung analisis dan pengembangan sistem akademik berbasis *Web Service*. Studi pustaka ini penting untuk menyeimbangkan pendekatan praktis dengan prinsip teoretis yang mendasari teknologi yang akan diimplementasikan.
- 6) Data Mata Kuliah: Informasi terkait mata kuliah yang diajarkan, kode mata kuliah, dan SKS (Satuan Kredit Semester).
- 7) Data Mahasiswa: Data pribadi mahasiswa, termasuk status akademik, IPK, dan data kontrak mahasiswa setiap semester.
- 8) Data Absensi: Data kehadiran mahasiswa selama perkuliahan berlangsung.
- 9) Data Berita Acara: Catatan resmi terkait jalannya proses perkuliahan yang berfungsi sebagai dokumentasi akademik.
- 10) Data Kontrak Mahasiswa: Kontrak perkuliahan yang diambil oleh mahasiswa setiap semester.
- 11) Data Survei: Survei terkait dengan evaluasi pengajaran, kepuasan mahasiswa, dan umpan balik dari mahasiswa untuk dosen.
- 12) Data Nilai: Rekapitulasi nilai dari setiap mata kuliah yang diambil oleh mahasiswa.
- 13) Data Rencana Perkuliahan Semester: Informasi tentang rencana pelaksanaan perkuliahan selama satu semester, termasuk silabus dan materi ajar.

### C. Kebutuhan Data

Tahapan awal yang dilakukan dalam pengembangan sistem akademik ini adalah pengumpulan data dengan menggunakan beberapa metode untuk memperoleh informasi yang akurat dan relevan. Langkah pertama adalah melakukan *interview* dengan Dr. Edy Saputra, M.Pd., Ketua Lembaga Penjaminan Mutu IAIN Takengon. Tujuan dari wawancara ini adalah untuk mendapatkan wawasan langsung mengenai proses dan kebutuhan utama dari sistem akademik di IAIN Takengon. Sebagai salah satu pihak yang memiliki pengetahuan mendalam terkait kualitas pendidikan dan kebutuhan akademik, masukan dari Dr. Edy sangat penting dalam merumuskan fitur dan fungsi utama dari sistem yang akan dibangun.

Langkah kedua adalah observasi langsung ke lokasi IAIN Takengon untuk melihat secara langsung kondisi sistem akademik yang berjalan. Observasi ini bertujuan untuk memahami dengan lebih baik bagaimana proses-proses akademik dijalankan, apa saja kendala yang dihadapi, serta mengidentifikasi terhadap alur kerja yang dapat ditingkatkan melalui sistem baru yang lebih efisien. Dalam observasi ini, data-data terkait dengan proses akademik dikumpulkan, termasuk absensi mahasiswa, berita acara perkuliahan, kontrak mahasiswa, survei kepuasan, serta rencana perkuliahan semester (RPS). Data-data ini sangat penting sebagai landasan dalam mengembangkan sistem yang mampu memenuhi kebutuhan akademik secara komprehensif.

Selain itu, proses pengumpulan data juga melibatkan analisis terhadap data-data lain yang relevan dengan sistem akademik. Data yang dikumpulkan meliputi:

- 1) Data User: Informasi pengguna sistem, termasuk mahasiswa, dosen, dan admin.
- 2) Data Fakultas: Struktur fakultas dan departemen yang ada di IAIN Takengon.
- 3) Data Program Studi: Informasi tentang program studi yang tersedia, mata kuliah, dan kurikulum.
- 4) Data Kelas: Pengelolaan kelas, jadwal perkuliahan, dan kapasitas kelas.
- 5) Data Dosen: Informasi mengenai dosen, seperti jadwal mengajar, bidang keahlian, dan riwayat pendidikan.

Semua data tersebut perlu dipertimbangkan dalam

proses perancangan sistem untuk memastikan bahwa sistem akademik yang dibangun dapat mengelola seluruh informasi dengan efisien, aman, dan mudah diakses oleh pengguna yang berwenang. Tujuan utama dari sistem ini adalah untuk meningkatkan efisiensi penyampaian informasi akademik, baik antara dosen, mahasiswa, maupun pihak administrasi, serta mendukung proses pengambilan keputusan berbasis data di institusi. Dengan demikian, integrasi data yang baik dan manajemen data yang terstruktur menjadi kunci utama keberhasilan sistem ini.

Sistem yang dirancang juga harus mampu memberikan informasi akademik secara real-time, mendukung kolaborasi antar fakultas, serta memfasilitasi pelaporan dan analisis kinerja akademik secara otomatis.

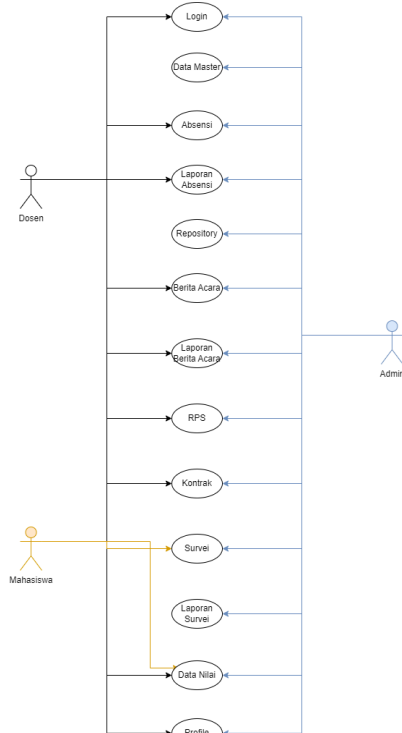
### D. Rancangan Sistem

Perancangan sistem dalam penelitian ini melibatkan beberapa tahap penting yang meliputi analisis kebutuhan, pemodelan diagram, serta perancangan arsitektur. Setiap tahap ini memiliki peran krusial dalam memastikan bahwa sistem yang dihasilkan tidak hanya memenuhi kebutuhan fungsional, seperti kemampuan untuk mengelola dan memproses data secara efisien, tetapi juga kebutuhan non-fungsional, seperti kinerja, keamanan, dan usability. Proses perancangan ini dilakukan dengan pendekatan yang sistematis dan terstruktur, bertujuan untuk menciptakan sistem yang efektif dan dapat diandalkan, sehingga dapat mendukung aktivitas akademik dengan baik. Dengan memperhatikan semua aspek ini, diharapkan sistem yang dikembangkan akan memberikan kemudahan dalam pengelolaan informasi, meningkatkan kolaborasi antar pengguna, dan mempercepat akses terhadap data yang dibutuhkan, sehingga mendukung tujuan utama dari pengembangan sistem informasi akademik. Selain itu,

perancangan sistem juga mempertimbangkan integrasi dengan sistem lain yang mungkin ada di lingkungan perguruan tinggi, agar komunikasi antar sistem dapat berlangsung dengan lancar dan efisien.

1) Use Case Diagram

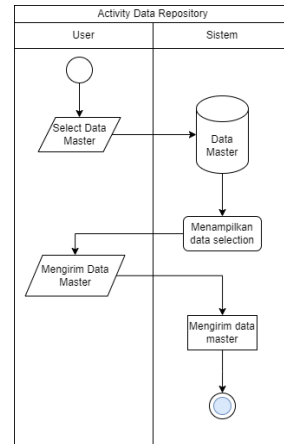
Perancangan diagram ini memiliki 3 aktor yaitu admin, dosen, dan mahasiswa. Admin dapat mengakses dan melakukan proses login, data master, absensi, laporan absensi, repository, berita acara, laporan berita acara, Rencana Perkuliahan Semester (RPS), kontrak, survei, laporan survei, nilai, dan profile. Mahasiswa hanya dapat mengakses survei dan nilai. Dosen dapat mengakses dan melakukan proses login, absensi, laporan absensi, berita acara, laporan berita acara, RPS, kontrak, nilai, dan profile. Use case diagram sistem ini diilustrasikan pada Gambar 1 berikut.



Gambar 1. Use Case Diagram

2) Activity Diagram

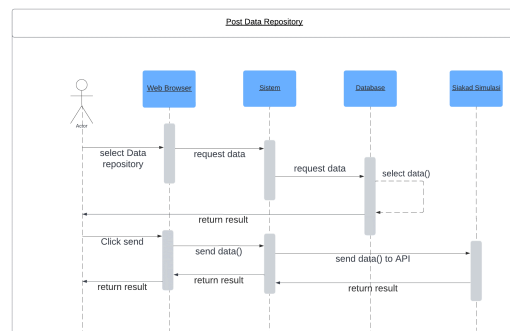
Activity Diagram adalah diagram yang memodelkan aliran aktivitas pada sistem. Adapun Activity Diagram pada sistem adalah sebagai berikut:



Gambar 2. Activity Diagram

3) Sequence Diagram

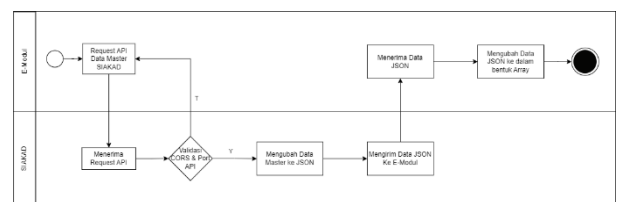
Sequence Diagram menggambarkan urutan waktu dari aksi dan pesan dalam sistem. Setiap pesan menandai komunikasi antar objek. Perancangan sequence diagram repository untuk sistem ini digambarkan seperti yang terdapat pada Gambar 3 berikut.



Gambar 3. Sequence Diagram

4) Arsitektur Web Service API

Arsitektur Web Service API dalam integrasi Sistem Informasi Akademik (SIKAD) IAIN Takengon dengan E-Modul digambarkan seperti Gambar 4 berikut.



Gambar 4. Sequence Diagram

Gambar 4 merupakan arsitektur web service API yang mempunyai beberapa proses diantaranya sebagai berikut.

- *Request API* Data Master Siakad. Sistem E-Modul IAIN akan mengirim permintaan data master dari SIAKAD. Data master tersebut meliputi: data fakultas, data program studi, data kelas, data mata kuliah, data dosen, dan data mahasiswa.
- Menerima *Request API*. SIAKAD akan menerima permintaan *API* dari E-Modul IAIN.
- Validasi *CORS* dan *Port API* SIAKAD akan memvalidasi akses *Cross-Origin Resource Sharing* dan *Port Hypertext Transfer Protocol* yang diberi akses oleh server, jika benar maka akan dilanjutkan untuk pengisian data jika tidak maka akan dikembalikan dengan membawa pesan.
- Mengubah Data Master ke *JSON* SIAKAD akan mengubah data master ke dalam format *JavaScript Object Notation*.
- Mengirim Data *JSON* ke E-Modul SIAKAD akan mengirim data *JSON* ke E-Modul Melalui *API*
- Menerima Data *JSON* E-Modul akan menerima data *JSON* dari SIAKAD Melalui *API*
- Mengubah Data *JSON* ke dalam bentuk *array* E-Modul akan mengubah *JSON* ke dalam bentuk *array* agar bisa di proses.

### III. HASIL DAN PEMBAHASAN

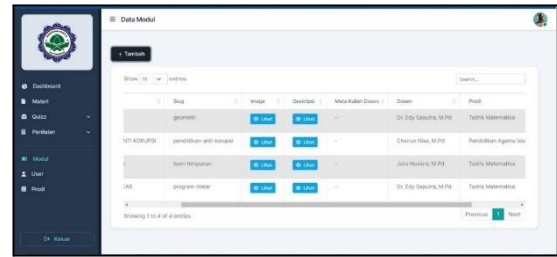
Bagian ini membahas hasil implementasi *API* pada SIAKAD IAIN Takengon serta hasil dan pengujian sistem yang telah dibangun. Hasil penelitian dan pengujian sistem ini akan dijelaskan secara lebih rinci sebagai berikut.

#### A. *API SIAKAD dan E-Modul*

Terdapat sebuah *route* yang menangani permintaan *HTTP GET* pada endpoint */matakuliah-dosen*. Ketika endpoint tersebut diakses, request akan diteruskan ke method *getData()* yang ada dalam *MataKuliahDosenController*. Dalam method *getData()*, dilakukan pengambilan data matakuliah dosens dari model *MataKuliahDosen* bersama dengan data terkait dosen dan mata kuliah yang masing-masing memiliki relasi dengan model tersebut. Data yang diambil tersebut kemudian dikonversi menjadi format *JSON* menggunakan method *json()* dan dikirimkan sebagai respons kembali kepada pengguna melalui fungsi *response()->json()*. Dengan demikian, ketika endpoint */matakuliah-dosen* diakses, pengguna akan menerima respons berupa data matakuliah dosens dalam format *JSON*.

Fungsi yang digunakan untuk mengambil data matakuliah dosen dari sebuah *API*. Pertama, fungsi melakukan permintaan *HTTP GET* ke *endpoint* yang ditentukan. Setelah itu, fungsi memeriksa apakah respons dari server berhasil atau tidak. Jika permintaan berhasil, data matakuliah dosens diekstrak dari respons dan dikembalikan dalam format *JSON*. Namun, jika permintaan tidak berhasil, fungsi akan mengembalikan sebuah *array* kosong.

Tampilan dari hasil pengiriman data mata kuliah dosen dari SIAKAD ke aplikasi E-Modul seperti pada Gambar 5 berikut.

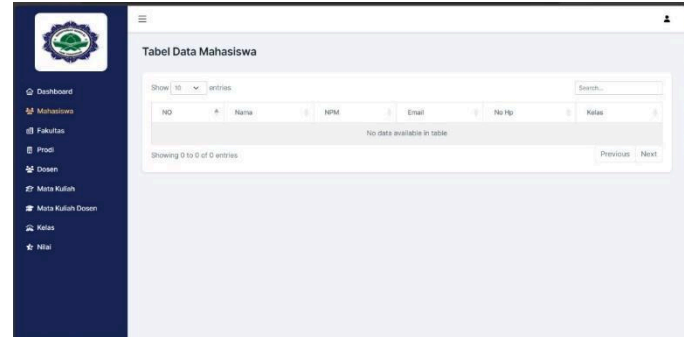


Gambar 5. Tampilan Halaman E-Modul

#### B. *API SIAKAD dan SIAKAD Simulasi*

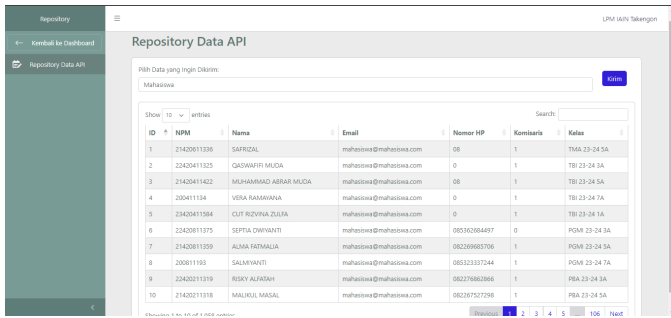
Implementasi dari sebuah end-point *API* dalam aplikasi simulasi Sistem Informasi Akademik (SIAKAD), yang bertugas menerima data dari luar dan menyimpannya ke dalam database sesuai dengan jenis data yang diterima. Saat diakses melalui metode *HTTP POST*, end-point akan mengambil data yang dikirimkan, memprosesnya berdasarkan jenis data ('data`Type`'), dan menyimpannya ke dalam database. Setiap jenis data diproses secara terpisah dan disimpan sesuai dengan model yang sesuai. Jika proses penyimpanan berhasil, end-point akan mengembalikan respons dengan status *HTTP 201 (Created)*, sementara jika terjadi kesalahan, respons akan berstatus *HTTP 500 (Internal Server Error)* beserta pesan kesalahan.

Gambar 6 berikut merupakan tampilan saat data masih kosong, di mana kolom-kolom seperti nama, NPM, email, nomor HP, dan kelas belum diisi.



Gambar 6. Tampilan Halaman Siakad Simulasi

Gambar 7 berikut merupakan tampilan untuk mengirim data berdasarkan pilihan dari formulir select. Formulir ini memungkinkan pengguna untuk memilih informasi yang ingin dikirim, seperti NPM (Nomor Pokok Mahasiswa), nama lengkap mahasiswa, alamat email, nomor telepon, nama komisaris, dan kelas yang diikuti. Data ini diambil dari Respository Data *API* yang diimplementasikan dalam Web Service berbasis *API Gateway*.

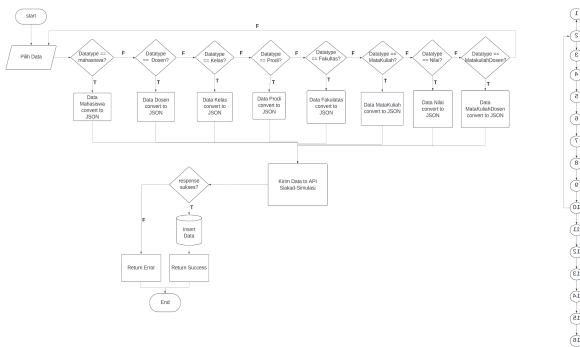


Gambar 7. Tampilan Halaman Repository API

C. Pengujian White Box

4) API SIAKAD – SIAKAD Simulasi

Gambar 8 menggambarkan perancangan flowchart dan flowgraph pengujian White Box API SIAKAD – SIAKAD Simulasi. Flowchart dan flowgraph pada Gambar 8 menjelaskan pengujian yang di mulai dari inputan data, perkondisian datatype == mahasiswa, dosen, kelas, prodi, fakultas, matakuliah, nilai dan matakuliahdosen, dilanjut dengan mengirim data ke siakad simulasi, perkondisian validasi data, merekam data ke database siakad simulasi dan return message. Kompleksitas siklomatis yang dijelaskan pada flowgraph diatas adalah 1 path dengan jalur: 1-2-3-4-5-6-7-8-9-10-11-12-13-14



Gambar 8. Flowchart & Flowgraph White Box API SIAKAD - SIAKAD Simulasi

Setelah ditemukan jalurnya, berikut pengujian test case dari API SIAKAD – SIAKAD Simulasi seperti Tabel 1.

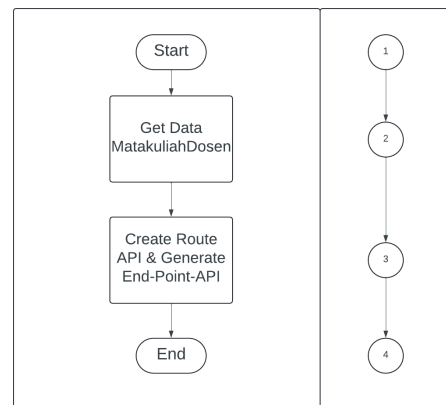
TABEL I  
HASIL PENGUJIAN WHITE BOX SIAKAD – SIAKAD SIMULASI

Path	Jalur	Skenario	Hasil Pengujian
1	1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16	1. Start 2. Input Data 3. Cek kondisi Datatype == mahasiswa 4. Cek kondisi Datatype == dosen 5. Cek kondisi Datatype == kelas	Benar (√) Salah (...)

6. Cek kondisi Datatype == prodi
7. Cek kondisi Datatype == fakultas
8. Cek kondisi Datatype == matakuliah
9. Cek kondisi Datatype == dosen
10. Cek kondisi Datatype == nilai
11. Cek kondisi Datatype == matakuliahdosen
12. Mengirim data ke endpoint siakad simulasi dan memasukkan data ke database
13. Cek kondisi response == sukses
14. Return success message
15. Return error message
16. End

5) API SIAKAD – E-Modul

Gambar 9 menggambarkan perancangan flowchart dan flowgraph pengujian White Box API SIAKAD – E-Modul. Flowchart dan flowgraph pada gambar 9 menjelaskan pengujian yang di mulai dari pengambilan data matakuliahdosen, dan membuat route API serta membuat end-point API. Kompleksitas siklomatis yang dijelaskan pada flowgraph diatas adalah 1 path dengan jalur : 1-2-3-4



Gambar 9. Flowchart & Flowgraph White Box API SIAKAD – E-Modul

Setelah ditemukan jalurnya, berikut pengujian test case dari API SIAKAD – E-Modul seperti Tabel 2.

TABEL II  
HASIL PENGUJIAN WHITE BOX SIAKAD – E-MODUL

Path	Jalur	Skenario	Hasil Pengujian
1	1-2-3-4	1. Start 2. Mengambil Data Matakuliahdosen 3. Membuat route API dan Membuat End-Point API	Benar (√) Salah (...)

#### D. Analisis Keseluruhan

Penerapan *Web Service berbasis API Gateway*, pada sistem ini mampu menyediakan layanan terintegrasi yang efisien bagi pengguna. Melalui *API Gateway*, SIAKAD dapat terhubung dengan berbagai sistem atau layanan lainnya secara seamless, memungkinkan pertukaran data yang cepat dan aman. Fitur-fitur seperti manajemen data master, manajemen berita acara dan absensi, serta survei dan manajemen soal dapat diakses dan diintegrasikan dengan sistem lain sesuai kebutuhan. Dengan demikian, pengguna tidak hanya mendapatkan akses yang komprehensif terhadap informasi akademik, tetapi juga memperoleh kemudahan dalam berinteraksi dengan sistem lain. Dukungan dari *Web Service berbasis API Gateway* memperkuat keandalan dan ketersediaan informasi dalam SIAKAD IAIN Takengon, menjadikannya solusi yang efektif dalam mendukung operasional institusi pendidikan tinggi tersebut.

#### IV. KESIMPULAN

Setelah melalui tahapan perancangan dan pengujian, dapat disimpulkan bahwa Sistem Informasi Akademik (SIAKAD) yang telah diintegrasikan dengan aplikasi E-Modul dan Simulasi-SIAKAD menggunakan *Web Service API* berhasil mengurangi redundansi data pada aplikasi E-Modul. Integrasi ini juga memfasilitasi simulasi SIAKAD dengan Forlap PDDIKTI, sehingga proses sinkronisasi data akademik antara SIAKAD dan Forlap menjadi lebih efisien. Dengan dukungan desain arsitektur yang baik, sistem SIAKAD yang terintegrasi mampu memenuhi berbagai kebutuhan akademik secara lebih terstruktur. Hal ini tidak hanya mendukung aktivitas perguruan tinggi, tetapi juga memperlancar pertukaran informasi antara sistem-sistem internal kampus maupun eksternal, termasuk dengan pihak terkait seperti Forlap PDDIKTI, sehingga informasi akademik dapat dikelola secara lebih optimal.

#### REFERENSI

- [1] Widodo, S., & Irawan, M., "Pengembangan Sistem Informasi Akademik Berbasis Web di Perguruan Tinggi," *Jurnal Teknologi Informasi dan Komputer*, vol. 6, no. 2, pp. 120-128, 2020.
- [2] Hidayat, M., & Syafriandi, "Analisis Implementasi SIAKAD Menggunakan Pendekatan *Web Service* di Perguruan Tinggi," *Jurnal Sistem Informasi*, vol. 15, no. 1, pp. 35-42, 2019.
- [3] Fadhillah, M., & Riyadi, D., "Implementasi *Web Service RESTful* untuk Integrasi Sistem Informasi di Lingkungan Perguruan Tinggi," *Jurnal Informatika*, vol. 8, no. 3, pp. 89-96, 2021.
- [4] Putra, A., & Anwar, T., "Penggunaan *Web Service API* pada Sistem Informasi Akademik untuk Integrasi Data Real-Time," *Journal of Information Systems*, vol. 11, no. 2, pp. 45-53, 2022.
- [5] Maulana, R., & Suryani, "Pengaruh Implementasi *Web Service* dalam Sistem Informasi Akademik Terhadap Efisiensi Pengelolaan Data," *Jurnal Teknologi dan Sistem Informasi*, vol. 10, no. 4, pp. 57-65, 2023.
- [6] Hasan, R., & Rahman, F., "A Comparative Study of REST and SOAP *Web Services* in Modern Application Development," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 4, pp. 25-34, 2021.
- [7] Singh, S., & Kumar, N., "Exploring the Use of REST and SOAP *Web Services* for Enterprise Applications," *Journal of Web Engineering*, vol. 10, no. 2, pp. 105-115, 2020.
- [8] Rahman, A., "Optimizing Academic Information Systems through RESTful *API Integration*," *Journal of Computer Science and Information Technology*, vol. 14, no. 3, pp. 200-210, 2022.
- [9] Kumar, P. (2019). *Real-Time Data Integration in Academic Information Systems Using REST APIs*. *Journal of Applied Computing*, 11(3), 189-200.
- [10] Gupta, R. & Ahmed, S. (2020). *Enhancing Decision-Making Processes in Universities Through *Web Services Integration**. *International Journal of Educational Technology*, 15(4), 322-334.