



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 5 No. 1 (2026) pp: 3121-3128

P-ISSN: 2963-9298, e-ISSN: 2963-914X

Pengembangan RESTful Web Service Untuk Integrasi Data Pada Sistem Informasi Akademik

Ikhwan

Sistem Informasi, Universitas Pamulang, Serang

dosen03358@unpam.ac.id

Abstrak

Salah satu aspek penting dalam perguruan tinggi adalah layanan sistem informasi kepada mahasiswa. Kualitas layanan sistem informasi menjadi perhatian utama karena berpengaruh terhadap tingkat kepercayaan dan kepuasan pengguna. Sistem akademik merupakan sistem yang digunakan untuk mendukung kegiatan belajar mengajar serta mempermudah pelayanan akademik kepada mahasiswa, seperti pengelolaan biodata, akses Kartu Rencana Studi (KRS) dan Kartu Hasil Studi (KHS). Namun, sistem akademik mengalami kendala dalam integrasi data dengan sistem lain akibat adanya perbedaan server, platform, maupun basis data yang digunakan. Hal ini menyebabkan pertukaran data menjadi kurang efisien dan berpotensi menimbulkan tidak konsistennya informasi. Oleh karena itu, diperlukan pengembangan RESTful Web Service berbasis Application Programming Interface (API) yang mampu mengintegrasikan data antara sistem akademik dengan sistem lain. Penelitian ini bertujuan untuk merancang dan membangun backend web service pada modul biodata mahasiswa, Kartu Rencana Studi (KRS), dan Kartu Hasil Studi (KHS) menggunakan bahasa pemrograman PHP. Metode pengembangan sistem yang digunakan adalah model waterfall, yang meliputi tahapan studi literatur, perancangan sistem, implementasi, dan pengujian. Hasil penelitian menunjukkan bahwa web service yang dikembangkan mampu menjembatani integrasi data lintas platform dan aplikasi. Pengujian sistem dilakukan menggunakan metode black box testing dengan bantuan tools Postman serta format pertukaran data berbasis JSON. Hasil pengujian menunjukkan bahwa seluruh endpoint API berjalan dengan baik dan sesuai dengan kebutuhan fungsional yang diinginkan.

Kata kunci : RESTful Web Service, Sistem Akademik, Integrasi Data, Pemrograman PHP

1. Pendahuluan

Seiring dengan perkembangan teknologi informasi yang semakin pesat, perguruan tinggi dituntut untuk meningkatkan mutu layanan akademik guna mendukung kualitas pendidikan. Peningkatan mutu layanan ini bertujuan untuk memenuhi kebutuhan mahasiswa serta meningkatkan kepercayaan dan kredibilitas institusi. Salah satu upaya yang dilakukan adalah dengan mengembangkan sistem akademik yang mampu mempermudah pengelolaan dan penyampaian informasi akademik kepada mahasiswa (Kaharu & Romadhona, 2026). Sistem akademik pada perguruan tinggi umumnya tidak berdiri sendiri, melainkan harus terintegrasi dengan sistem lain seperti sistem keuangan, sistem alumni, dan sistem pendukung lainnya. Informasi yang dikelola dalam sistem akademik meliputi biodata mahasiswa, Kartu Rencana Studi (KRS), dan Kartu Hasil Studi (KHS). Oleh karena itu, diperlukan mekanisme pertukaran data yang aman, fleksibel, dan mudah dikembangkan agar integrasi antar sistem dapat berjalan dengan baik.

Salah satu solusi yang banyak digunakan untuk mendukung integrasi antar sistem adalah penggunaan *web service*. *Web service* merupakan standar pertukaran data yang berfungsi sebagai penghubung antara *frontend* dan *backend* serta memungkinkan komunikasi antar aplikasi lintas platform (Novianto & Munir, 2022) dan (Grasias & Supriyadi, 2022). RESTful API (*Representational State Transfer Application Programming Interface*) merupakan salah satu pendekatan *web service* yang banyak digunakan karena mendukung skalabilitas, keamanan, serta performa yang baik dengan metode *request* seperti GET, POST, PUT, dan DELETE (Aziz et al., 2025) dan (Shofyan & Shita, 2024). Dengan API ini pertukaran data menjadi lebih cepat dan *real time* (Pangestu et al., 2022). Pada kondisi saat ini, universitas belum memiliki *web service* yang dapat memfasilitasi integrasi data sistem akademik, khususnya pada modul KRS dan KHS. Padahal, kedua modul tersebut merupakan komponen penting dalam sistem akademik. Modul KRS berfungsi untuk mengelola kontrak mata kuliah yang diambil oleh mahasiswa, sedangkan modul KHS menyajikan hasil studi yang telah dicapai mahasiswa selama proses pembelajaran.

Beberapa penelitian sebelumnya telah membahas implementasi RESTful API pada berbagai sistem. Rahayu dkk (2025) mengimplementasikan REST API untuk pencatatan akta kelahiran dan kematian berbasis web dengan pemodelan arsitektur menggunakan UML (Rahayu et al., 2025). Penelitian lain menerapkan REST API pada sistem sumber daya manusia berbasis web menggunakan framework Laravel dengan metode pengembangan *waterfall* (Simbulan & Aryanto, 2024) dan (Hakim et al., 2025). Selain itu, penelitian oleh Muharom dkk (2025) menjelaskan penggunaan RESTful API untuk komunikasi antar sistem dengan pengujian menggunakan *tools Postman* melalui metode HTTP GET, POST, PUT, dan DELETE (Muharom et al., 2025).

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk membangun *backend web service* berbasis RESTful API pada modul Kartu Rencana Studi (KRS) dan Kartu Hasil Studi (KHS) mahasiswa. Pengembangan sistem dilakukan menggunakan metode *waterfall* yang meliputi tahapan analisis studi literatur, perancangan sistem, implementasi, dan pengujian sistem. Pengujian sistem dilakukan menggunakan metode *black box testing* dengan bantuan *tools Postman* untuk memastikan fungsionalitas API berjalan sesuai dengan kebutuhan. *Web service* yang dikembangkan menggunakan format pertukaran data berbasis JSON (*JavaScript Object Notation*), yang bersifat ringan, mudah dibaca, dan mendukung pertukaran data lintas aplikasi (Saputra & Herdiyatomoko, 2024) dan (Kurniawati et al., 2025). Bahasa pemrograman apapun tidak akan bergantung pada format JSON, karena JSON merupakan gaya bahasa yang umum digunakan (Fadillah & Khairullah, 2021). Restfull API ini memiliki peran yang sangat penting untuk menghubungkan beberapa sistem, lintas platform atau aplikasi. RESTful bekerja dengan cara *resource-oriented*, pada restful *web services* client mengakses services atau data yang ada di web server (Gustiegan & Painem, 2022).

Untuk menguji hasil pengembangan RESTful API dan *deployment* dilakukan *black box testing* untuk memastikan bahwa API yang telah dibuat dapat berjalan dengan baik (Chandra et al., 2024). Pengujian sistem dilakukan dengan metode *Blackbox Testing* yang dilakukan menggunakan *web service test* pada *tools postman* dengan pertukaran data dalam bentuk format JSON. *Blackbox* merupakan cara yang dapat digunakan untuk mencari kesalahan dan menguji fungsi aplikasi apakah sudah berjalan dengan baik, parameter yang diterima dan *output* yang dihasilkan sudah sesuai dengan yang diinginkan (Ardiansyah & Kurniasih, 2019). Penelitian ini diharapkan dapat menghasilkan *backend web service* yang mampu menjembatani integrasi data lintas platform dan aplikasi, sehingga mendukung pengembangan sistem akademik yang lebih terstruktur.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan metode kualitatif yang bertujuan untuk mendeskripsikan permasalahan yang terjadi pada sistem akademik, khususnya pada kebutuhan integrasi data modul Kartu Rencana Studi (KRS) dan Kartu Hasil Studi (KHS). Pendekatan kualitatif digunakan untuk menganalisis kondisi sistem yang berjalan serta menentukan kebutuhan pengembangan *backend web service*. Analisis kebutuhan dibutuhkan untuk mengetahui apa saja yang diperlukan dalam proses pengembangan *web service*. Proses ini dilakukan dengan pengamatan langsung pada modul aplikasi, agar nantinya dapat diintegrasikan dengan sistem atau aplikasi lain (Novianto & Munir, 2022). Berdasarkan hasil analisa yang telah dilakukan, didapat hasil pada table berikut.

Tabel 1. Analisa Kebutuhan

No	Rancangan	Kondisi
1	Database KRS dan KHS	Tersedia
2	Framework <i>Web Service</i> pada sistem akademik	Belum Tersedia
3	Modul data transfer	Sudah ditentukan
4	Format response data	Sudah ditentukan
5	Dokumentasi	Belum tersedia

Pengumpulan data dilakukan melalui observasi langsung terhadap sistem yang diteliti, dengan mengidentifikasi struktur sistem, proses, serta alur interaksi antara pengguna dan sistem. Selain itu, studi kepustakaan dilakukan dengan mengkaji artikel dan jurnal ilmiah yang relevan guna mendukung landasan teori dan penelitian terdahulu. Proses sistem yang berjalan saat ini dapat dilihat pada gambar dibawah ini.

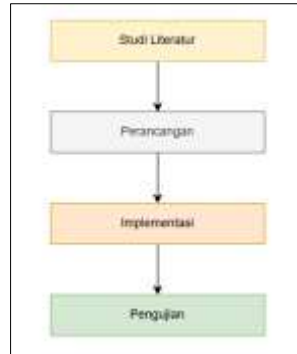


DOI: <https://doi.org/10.31004/riggs.v5i1.6547>

Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Gambar 1. Skema Sistem Informasi Akademik

Metode pengembangan sistem yang digunakan dalam perancangan *backend web service* adalah model *waterfall*. Model ini dipilih karena memiliki tahapan yang sistematis dan terstruktur. Tahapan dalam metode *waterfall* meliputi studi literatur, perancangan sistem, implementasi, dan pengujian sistem. Pengujian sistem dilakukan untuk memastikan bahwa RESTful API yang dikembangkan telah berjalan sesuai dengan kebutuhan fungsional yang ditetapkan. Model *waterfall* dapat dilihat pada gambar di bawah ini.



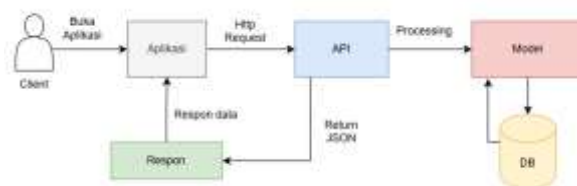
Gambar 2. Metode Waterfall

Tahap pertama dalam penelitian ini adalah studi literatur yang digunakan sebagai acuan dalam pengembangan penelitian. Referensi yang digunakan diperoleh dari jurnal dan penelitian terdahulu yang relevan dengan topik *web service* dan RESTful API. Studi literatur ini bertujuan untuk memperkuat landasan teori serta memahami pendekatan yang telah dilakukan pada penelitian sebelumnya. Tahap selanjutnya adalah perancangan sistem, yang dilakukan dengan menganalisis kebutuhan pada *web service* yang akan dibangun. Pada tahap ini dilakukan perancangan sistem *web service* dan modul – modul yang akan dibuat. Modul tersebut terdiri dari, modul profil mahasiswa, Kartu Rencana Studi (KRS) dan Kartu Hasil Studi (KHS). Selanjutnya dilakukan desain *system deployment* untuk mengetahui kerangka yang akan dibuat pada *web service* pada penelitian ini.

Tabel 2. Kerangka System Deployment

No	Kebutuhan Modul	Deskripsi	Endpoint
1	Modul Mahasiswa	Berfungsi untuk mengambil data mahasiswa	/mahasiswa/{nim}
2	Modul KRS	Berfungsi untuk mengambil data Kartu Rencana Studi Mahasiswa (KRS)	/krs/{nim}
3	Modul KHS	Berfungsi untuk mengambil data Kartu Hasil Studi (KHS)	/khs/{nim}

Tahap implementasi dilakukan berdasarkan hasil perancangan sistem yang telah dibuat. Arsitektur sistem dirancang menggunakan model RESTful API dan dikembangkan menggunakan bahasa pemrograman PHP dengan format pertukaran data berbasis JSON. Model *web service* dapat dilihat pada gambar 3 di bawah ini.



Gambar 3. Model Web Service

Tahap terakhir adalah pengujian sistem, yang bertujuan untuk memastikan bahwa *web service* yang dikembangkan telah berjalan sesuai dengan kebutuhan fungsional. Metode pengujian yang digunakan adalah *black box testing* dengan bantuan *tools Postman*. Metode ini memungkinkan untuk mengevaluasi secara fungsionalitas *Web service* yang dibuat tanpa melihat struktur kode secara langsung (Hakim et al., 2025). Pengujian difokuskan pada fungsional RESTful *web service* dengan menjalankan metode http GET, POST, PUT, dan DELETE melalui REST client untuk memastikan setiap layanan API berfungsi dengan baik. Langkah pengujian dilakukan untuk mengetahui apakah RESTFull *Web Service* sudah layak untuk digunakan.

Tabel 3. Rancangan Blackbox Testing

No	Method	Deskripsi	Response data	Keterangan
1	GET	Ambil data dari Web Service	200/500	Berhasil/gagal
2	POST	Tambah data dari Web Service	200/500	Berhasil/gagal
3	PUT	Update data dari Web Service	200/500	Berhasil/gagal
4	DELETE	Hapus data dari Web Service	200/500	Berhasil/gagal

Menguji fungsionalitas fitur-fitur yang ada dengan menggunakan *blackbox* testing dan Postman, serta berfokus pada pengujian GET, POST, PUT, DELETE pada *web service*. Pengujian ini bertujuan untuk mengetahui apakah *web service* yang dibangun sudah berjalan sesuai dengan yang diharapkan.

3. Hasil dan diskusi

Bagian ini membahas hasil dari implementasi RESTful API yang dibuat menggunakan Bahasa pemrograman PHP yang berfokus pada sisi *backend*. Hal yang akan dibahas pada bagian ini yaitu implementasi kode program dan hasil pengujian.

3.1 Arsitektur Sistem

Penelitian ini hanya berfokus pada sisi *backend* yaitu *web service* dengan model RESTful API. Pada sisi *client* dapat melakukan proses *read*, *insert*, *update* dan *delete* data. Gambaran proses pertukaran data dapat dilihat pada penjelasan dibawah ini.

- Client* akan melakukan *http request* melalui *endpoint* dengan menggunakan fungsi GET, POST, PUT dan DELETE.
- Server* menerima permintaan tersebut dan memproses sesuai dengan data yang dibutuhkan oleh *client*. Setelah permintaan valid, maka selanjutnya *server* mengembalikan data atau response ke *client* dalam bentuk format JSON.
- Client* menerima data respon dari *server* dalam bentuk JSON sesuai yang diinginkan.

Endpoint terdiri dari profil, KRS dan KHS mahasiswa. Berikut ini adalah format *endpoint* yang dibangun pada penelitian ini.

Tabel 4. Endpoint

No	Method	Profile Mahasiswa	KRS	KHS
1	GET	/api/mahasiswa/{nim}	/api/krs/{nim}	/api/khs/{nim}
2	POST	/api/mahasiswa/	/api/krs/	/api/khs/
3	PUT	/api/mahasiswa/{nim}	/api/krs/{nim}	/api/khs/{nim}
4	DELETE	/api/mahasiswa/{nim}	/api/krs/{nim}	/api/khs/{nim}

3.2 Implementasi Kode Program

Pengembangan RESTful API menggunakan framework CodeIgniter versi 4. Pola struktur pada framework ini menggunakan MVC yaitu Model-View-Controller. Controller dan Model digunakan untuk pengembangan *web service* ini. Dalam penelitian ini Controller dan Model yang digunakan yaitu dengan komponen *namespace App\Controllers\Api* dan *namespace App\Models*. Selanjutnya masing - masing controller tersebut dibuat sebuah *function* pada setiap metode yang terdiri dari *function index*, *show*, *create*, *update* dan *delete*. Apabila data yang dikirim valid dan sesuai dengan format JSON yang ditentukan, maka sistem akan mengirimkan respon data 200. Jika sebaliknya, data yang dikirim tidak valid maka sistem akan memberikan respon data 500 atau *Internal server error*. *Function* pada setiap metode pada pemrograman PHP yang digunakan pada penelitian ini dapat dilihat pada gambar dibawah ini.

```

public function show($nim = null)
{
    $data = $this->model->find($nim);
    if (!$data) {
        return $this->failNotFound('Data tidak ditemukan');
    }
    return $this->respond($data);
}

```

Gambar 4. *function GET Mahasiswa*

```

public function update($nim = null)
{
    $data = $this->request->getJSON(true);
    if (!$this->model->find($nim)) {
        return $this->failNotFound('Data tidak ditemukan');
    }

    $this->model->update($nim, $data);
    return $this->respond([
        'status' => true,
        'message' => 'Data berhasil diupdate'
    ]);
}

```

Gambar 5. *function PUT*

```

public function show($nim)
{
    $data = $this->krsModel->where('NIM', $nim)->findAll();
    if (empty($data)) {
        return $this->response->setStatusCode(404)->setJSON([
            'status' => false,
            'message' => 'Data KRS tidak ditemukan'
        ]);
    }

    return $this->response->setJSON([
        'status' => true,
        'data' => $data
    ]);
}

```

Gambar 6. *function GET KRS*

```

public function show($nim)
{
    $data = $this->khsModel->where('NIM', $nim)->findAll();
    if (empty($data)) {
        return $this->response->setStatusCode(404)->setJSON([
            'status' => false,
            'message' => 'Data KHS tidak ditemukan'
        ]);
    }

    return $this->response->setJSON([
        'status' => true,
        'data' => $data
    ]);
}

```

Gambar 7. *function GET KHS*

3.2. Hasil Pengujian

Web Service di uji berdasarkan fungsionalitas dari API yang dibangun pada penelitian ini yang meliputi pengujian dengan model format *endpoint Web Service*. Pada proses ini *web service* dapat diuji dengan RESTFull (*Representational State Transfer*) yaitu *request client* yang dilakukan melalui *http request*. Pengujian fungsional dilakukan dengan menggunakan *tools postman*, endpoint yang akan diuji yaitu pada modul profil Mahasiswa, Kartu Rencana Studi (KRS) dan Kartu Hasil Studi (KHS).

Hasil pengujian dengan postman terdiri dari fungsi GET, POST, PUT dan DELETE. Dari masing – masing *endpoint* tersebut akan memiliki fungsi yang berbeda. Fungsi GET akan mengambil semua data atau hanya data dengan nim tertentu, sedangkan metode POST digunakan untuk memasukan data sesuai dengan kolom – kolom yang ada pada tabel *database* dan untuk memasukan data tersebut dibutuhkan format JSON sebagai paramater dan komponen body RAW yang berisi data parameter yang ingin dimasukan ke dalam tabel *database* dengan *postman*. Fungsi PUT digunakan untuk merubah data yang sudah ada berdasarkan kriteria tertentu seperti pada nim mahasiswa. Untuk fungsi DELETE digunakan untuk menghapus data dengan kriteria tertentu, fungsi DELETE harus digunakan dengan hati – hati karena akan menghapus data dengan permanen. Dari semua masing – masing fungsi tersebut akan mendapatkan format pertukaran data dalam bentuk JSON yang berisi nilai 200 jika berhasil dan 500 jika gagal. Hasil pengujian dengan postman dapat dilihat pada gambar di bawah ini.



Gambar 8. Pengujian modul Profil Mahasiswa dengan GET



Gambar 9. Pengujian modul KRS dengan POST



Gambar 10. Pengujian modul KRS dengan GET



Gambar 11. Pengujian modul KHS dengan POST



Gambar 12. Pengujian modul KHS dengan GET

Setelah dilakukan pengujian dengan request parameter nim yang berisi nilai 202011001 didapat hasil yang dikembalikan oleh server yaitu status *true* yang artinya data berhasil ditemukan dan kemudian ditampilkan. Status kode yaitu 200 yang artinya permintaan yang dikirim dapat diterima dan diproses oleh server.

3.3 Black box testing

Pengujian sistem dilakukan dengan *blackbox testing* yaitu pengujian yang fokus pada fungsionalitas sistem. Tujuan dari metode ini yaitu ingin memastikan *output* sudah sesuai dengan yang diharapkan. Hasil pengujian dapat dilihat pada tabel 5 dibawah ini.

Tabel 5. Hasil Pengujian

No	Endpoint	Method	Kasus Test	Response	Hasil Pengujian
1	/api/mahasiswa/	GET	Mengirim dan menampilkan semua data yang sesuai format JSON	200 Ok	Berhasil
2	/api/mahasiswa/202011001	GET	Mengirim data berdasarkan nim yang sesuai format JSON	200 Ok	Berhasil
3	/mahasiswa/	POST	Mengirim data yang sesuai format JSON	200 Ok	Berhasil
4	/mahasiswa/	POST	Mengirim data yang tidak sesuai	500 Internal Server Error	Gagal
5	/api/mahasiswa/202011001	PUT	Ubah data yang sesuai dengan format JSON	200 Ok	Berhasil
6	/api/mahasiswa/202011001	DELETE	Menghapus data yang sesuai format JSON	200 Ok	Berhasil
7	/api/krs/	GET	Mengirim dan menampilkan semua data yang sesuai format JSON	200 Ok	Berhasil
8	/api/krs/202011001	GET	Mengirim data berdasarkan nim yang sesuai format JSON	200 Ok	Berhasil
9	/api/krs/	POST	Mengirim data yang sesuai format JSON	200 Ok	Berhasil
10	/api/krs/	POST	Mengirim data yang tidak sesuai	500 Internal Server Error	Gagal

11	/api/krs/202011001	PUT	Ubah data yang sesuai dengan format JSON	200 Ok	Berhasil
12	/api/krs/202011001	DELETE	Menghapus data yang sesuai format JSON	200 Ok	Berhasil
13	/api/khs/	GET	Mengirim dan menampilkan semua data yang sesuai format JSON	200 Ok	Berhasil
14	/api/khs/202011001	GET	Mengirim data berdasarkan nim yang sesuai format JSON	200 Ok	Berhasil
15	/api/khs/	POST	Mengirim data yang sesuai format JSON	200 Ok	Berhasil
16	/api/khs/	POST	Mengirim data yang tidak sesuai	500 Internal Server Error	Gagal
17	/api/khs/202011001	PUT	Ubah data yang sesuai dengan format JSON	200 Ok	Berhasil
18	/api/khs/202011001	DELETE	Menghapus data yang sesuai format JSON	200 Ok	Berhasil

Dari hasil pengujian *tools postman* dan teknik *blackbox testing* menunjukkan sistem *web service* sudah berjalan dengan baik dan sesuai dengan yang diharapkan.

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa *web service* pada sistem akademik berhasil dikembangkan menggunakan bahasa pemrograman PHP dengan arsitektur RESTful API. *Web service* ini dirancang untuk memudahkan pengembang sistem maupun aplikasi lain dapat melakukan integrasi data dengan sistem akademik secara efektif dan efisien. Pengembangan RESTful *web service* dilakukan menggunakan model *waterfall*, yang meliputi tahapan studi literatur, perancangan, implementasi, dan pengujian sistem. Modul yang berhasil dikembangkan pada penelitian ini meliputi profil mahasiswa, Kartu Rencana Studi (KRS), dan Kartu Hasil Studi (KHS). Seluruh layanan API telah diimplementasikan menggunakan metode http GET, POST, PUT, dan DELETE. Hasil pengujian menunjukkan bahwa REST server mampu menghasilkan response data dalam format JSON serta menyediakan layanan yang diminta oleh client dengan cepat dan akurat. Pengujian fungsional dilakukan menggunakan metode *black box testing* dengan bantuan *tools Postman*, dan hasilnya menunjukkan bahwa seluruh fitur API dapat berjalan dengan baik serta memenuhi kebutuhan sistem yang telah ditetapkan. Pengujian *response time* pada REST Server juga telah dilakukan menggunakan *tools Postman*, dan hasil pengujian menunjukkan bahwa sistem memiliki performa yang baik dalam melayani permintaan data dari client. Dengan demikian, dapat disimpulkan bahwa *web service* yang dikembangkan telah berfungsi sesuai dengan tujuan penelitian. Sebagai pengembangan ke depan, diharapkan sistem *web service* ini dapat dilengkapi dengan mekanisme keamanan yang lebih kompleks, seperti penggunaan token API atau metode autentikasi yang lebih aman, guna meningkatkan tingkat keamanan akses data pada sistem akademik.

Referensi

- Ardiansyah, A., & Kurniasih, M. (2019). Implementasi Algoritma AES-256 Untuk Pengamanan Layanan API Pada Restful Dengan Autentikasi Json Web Tokens. *Seminar Nasional Inovasi Teknologi – SNITek 2019*, 315–327.
- Aziz, F. A., Subandri, M. A., & Armada. (2025). Penerapan REST API pada Aplikasi Antarmuka Alat Pemantauan Tambak Udang. *JOURNAL OF APPLIED COMPUTER SCIENCE AND TECHNOLOGY (JACOST)*, 6(2), 82–91. <https://doi.org/https://doi.org/10.52158/19qkf837>
- Chandra, C., Wijaya, F., Gunawan, J. A., Lee, J. R., & Maulana, A. (2024). Perancangan dan Implementasi RESTful API untuk Aplikasi Mobile Pembelajaran Flora dan Fauna pada Google Cloud Platform. *Jurnal Sains Teknologi Dan Sistem Informasi*, 4(1), 58–69. <https://doi.org/10.54259/satesi.v4i1.2850>
- Fadillah, A., & Khairullah. (2021). Application of the Codeigniter Framework and Webservice on the Application for Personnel Information at the University of Muhammadiyah Bengkulu. *Jurnal KOMITEK*, 1(2), 418–425. <https://doi.org/https://doi.org/10.53697/jkomitek.v1i2>
- Grasias, N. J., & Supriyadi. (2022). Perancangan REST Web Service Untuk Pengembangan Sistem Pengajuan Simpan Pinjam PT. XYZ. *Jurnal Teknologi Informatika Dan Komputer MH. Thamrin*, 8(2), 72–86. <https://doi.org/https://doi.org/10.37012/jtik.v8i2.1182>
- Gustiegan, G. Y., & Painem. (2022). IMPLEMENTASI WEB SERVICE RESTFUL DENGAN AUTENTIKASI JSON WEB TOKEN DAN ALGORITMA KRIPTOGRAFI AES-256 UNTUK APLIKASI PEMINJAMAN LABORATORIUM BERBASIS IMPLEMENTATION OF A RESTFUL WEB SERVICE WITH JSON WEB TOKEN AUTHENTICATION AND AES-256 CRYPTOGRAPHIC ALGOR. *Bit (Fakultas Teknologi Informasi Universitas Budi Luhur)*, 19(1).
- Hakim, A. A., Utomo, D., Putro, R., Pramudyo, L. D., Syah, K., Mayya, K., Aditya, M., Aryanda, F., & Akbar, F. A. (2025). Implementasi RESTful API Web Wedding Organizer Ruang Hati menggunakan Laravel dan VueJs. *Seminar Nasional Informatika Bela Negara (SANTIKA)*, 5, 328–342.
- Kaharu, N. A., & Romadhona, A. (2026). Implementasi Konsep Object-Oriented Programming dalam Pengembangan Rest Api Menggunakan Framework Laravel. *Journal of Artificial Intelligence and Digital Business (RIGGS)*, 4(4), 10457–10465. <https://doi.org/https://doi.org/10.31004/riggs.v4i4.5165>

DOI: <https://doi.org/10.31004/riggs.v5i1.6547>

Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

9. Kurniawati, D., Kusjani, A., Buwono, R. C., & Ridhoni, M. A. (2025). Pengembangan Sistem Transformasi dan Konversi Data Berbasis Web Menggunakan Arsitektur RESTful API. *BULLETIN OF COMPUTER SCIENCE RESEARCH*, 5(6), 1395–1402. <https://doi.org/10.47065/bulletincsr.v5i6.818>
10. Muharom, L. A., Permata, A. D., & Oktavianto, H. (2025). Implementasi Web Service Restful API Pada Modul Wisata Aplikasi Malldesa Implementation of Restful API Web Service in the Tourism Module of the Malldesa Application. *Jurnal Sistem Dan Teknologi Informasi Indonesia*, 10(2), 146–154. <https://doi.org/https://doi.org/10.32528/justindo.v10i2.3703>
11. Novianto, M. A., & Munir, S. (2022). ANALISIS DAN IMPLEMENTASI RESTFUL API GUNA PENGEMBANGAN SISTEM INFORMASI AKADEMIK PADA PERGURUAN TINGGI. *Jurnal Informatika Terpadu*, 8(1), 47–61.
12. Pangestu, H. A., Kurniadi, D., & Septiana, Y. (2022). Aplikasi Pengelolaan Data Pegawai Berbasis REST API untuk Transfer Data Real Time dengan Framework Codeigniter. *Jurnal Algoritma*, 19(1), 313–322. <https://doi.org/https://doi.org/10.33364/algoritma/v.19-1.1090>
13. Rahayu, S., Sutedi, A., & Mutiara, C. (2025). Implementasi Rest Api Untuk Pencatatan Akta Kelahiran dan Kematian Pada Platform Web. *Jurnal Algoritma*, 22(1), 336–346. <https://doi.org/10.33364/algoritma/v.22-1.1704>
14. Saputra, L. S., & Herdiyatomoko, H. F. (2024). IMPLEMENTASI REST API PADA SISTEM PENDATAAN KAPAL MENGGUNAKAN LARAVEL 9 DAN NUXTJS. *Journal Computer and Technology*, 2(1), 10–21. <https://doi.org/https://doi.org/10.69916/comtechno.v2i1.138>
15. Shofyan, F., & Shita, R. T. (2024). Implementasi Web Service Restful API dengan Autentikasi Personal Access Tokens dan Algoritma AES 256. *Jurnal TICOM: Technology of Information and Communication*, 12, 108–114.
16. Simbulan, R., & Aryanto, J. (2024). Jurnal Indonesia : Manajemen Informatika dan Komunikasi Implementasi REST API Web Services pada Aplikasi Sumber Abstrak Jurnal Indonesia : Manajemen Informatika dan Komunikasi. *Jurnal Indonesia : Manajemen Informatika Dan Komunikasi*, 5(1), 552–560. <https://doi.org/https://doi.org/10.35870/jimik.v5i1.511>