

# Perancangan REST API Untuk Pengembangan Sistem SPBE di Desa Tembok

Rivan Nurdin<sup>1</sup>, Rangga Sanjaya<sup>2</sup>

<sup>1</sup>Program Studi Teknik Informatika, Universitas Adhirajasa Reswara Sanjaya

<sup>2</sup>Program Studi Sistem Informasi, Universitas Adhirajasa Reswara Sanjaya

e-mail: <sup>1</sup>rivannurdin001@gmail.com, <sup>2</sup>rangga@ars.ac.id

## Abstrak

Penelitian ini menyoroiti peran Sistem Pemerintahan Berbasis Elektronik (SPBE) dengan RST API dalam meningkatkan efisiensi pemerintahan desa. Pendekatan sistem informasi berbasis API di Desa Tembok, dengan metode waterfall dan desain REST API, terbukti efektif dalam menciptakan solusi berkelanjutan. Proses sekuensial memastikan tahap pengembangan dilakukan dengan baik. Desain API REST menjadi kunci dalam mempermudah komunikasi antar aplikasi, meningkatkan aksesibilitas dan efisiensi SPBE. Implementasi ini berpotensi menciptakan dasar inovasi teknologi di masa depan, memberikan manfaat signifikan bagi masyarakat lokal, dan memenuhi standar regulasi pemerintahan

**Kata kunci**—Perancangan, REST API, Sistem Pemerintah Berbasis Elektronik (SPBE)

## Abstract

*This research highlights the role of Electronic Based Government Systems (SPBE) with REST API in increasing the efficiency of village government. The API-based information system approach in Wall Village, with the waterfall method and REST API design, has proven effective in creating sustainable solutions. The sequential process ensures that the development stages are carried out well. REST API design is key in simplifying communication between applications, increasing SPBE accessibility and efficiency. This implementation has the potential to create the basis for future technological innovation, provide significant benefits to local communities, and meet government regulatory standards*

**Keywords**—Planning, REST API, Electronic-Based Government System

---

**Corresponding Author:**

**Rangga Sanjaya,**

Email: rangga@ars.ac.id

---

## 1. PENDAHULUAN

Digitalisasi pada bidang pemerintahan telah menjadi salah satu tren penting dalam upaya meningkatkan efisiensi, transparansi, dan akuntabilitas pemerintah di seluruh dunia. Digitalisasi ini mencakup penggunaan teknologi informasi, komunikasi, dan internet untuk mengotomatiskan proses administratif, mempermudah akses publik terhadap layanan pemerintah, serta meningkatkan interaksi antara pemerintah dan masyarakat. Melihat potensi teknologi informasi sebagai jembatan birokrasi baru yang dapat dimanfaatkan untuk kepentingan masyarakat dan negara, Pemerintah Republik Indonesia menerbitkan UU Nomor 11 Tahun 2008 dan UU Nomor 25 Tahun 2009 yang secara seragam menjelaskan tentang penyelenggaraan Sistem Pemerintahan Berbasis Elektronik (SPBE) dalam lingkup Daerah dan Nasional untuk kepentingan pelayanan publik.

Dengan diberlakukannya Undang-Undang Nomor 6 Tahun 2014 tentang desa, pemerintah desa memiliki kewenangan dalam membangun infrastruktur secara mandiri, maka dari itu pemerintah desa tembak berinisiatif untuk merancang dan mengimplementasikan SPBE yang dapat digunakan oleh masyarakat di wilayah tersebut. Dalam implementasinya perancangan SPBE membutuhkan desain infrastruktur backend yang mumpuni untuk mendukung proses pengembangan. Sistem backend yang dimaksud harus memiliki kemampuan untuk menyesuaikan diri dengan perubahan-perubahan yang mungkin terjadi dalam proses bisnis yang dikelolanya [1]. Peneliti berpandangan bahwa hal ini dapat dicapai dengan menerapkan konsep sistem berbasis *Application Programming Interface* (API) dengan menggunakan arsitektur *Representational State Transfer* (REST) dalam pengembangan bagian belakang SPBE.

Sistem informasi yang berdasarkan pada konsep API memungkinkan penggunaan yang lebih luas karena pemisahan antara logika di dalam sistem dengan logika di dalam antarmuka. Sistem berbasis API juga mempermudah kerjasama dengan sistem lain. Ini akan sangat berguna jika dalam waktu yang akan datang, proses bisnis yang dikelola memerlukan interaksi dengan sistem lain, atau setidaknya akan menjadi sangat bermanfaat saat mengembangkan modul baru dalam sistem informasi terkait [2].

Sistem Pemerintah Berbasis Elektronik (SPBE) merupakan pendekatan yang inovatif dalam administrasi pemerintahan yang mengintegrasikan teknologi informasi dan komunikasi untuk menyediakan layanan pemerintahan yang lebih efisien, accessible, dan transparan. SPBE mencakup penerapan teknologi digital dalam pengelolaan data pemerintah, pelayanan publik, pengembangan aplikasi berbasis web, dan sistem perizinan online [3]. Dalam konteks pemerintah desa, penerapan SPBE melibatkan tahap studi kelayakan, perencanaan proyek, memastikan infrastruktur, pengembangan aplikasi SPBE, dan pelatihan staf [4].

Penerapan REST API, IBM Cloud Education [5] juga mengemukakan bagaimana cara kerja dari REST API. REST API berkomunikasi melalui permintaan HTTP untuk melakukan fungsi-fungsi standar pada basis data, seperti pembuatan, pembacaan, pembaruan, dan penghapusan catatan (dikenal sebagai CRUD) dalam suatu sumber daya. Sebagai contoh, REST API menggunakan permintaan GET untuk mengambil catatan, POST untuk membuat catatan baru, PUT untuk memperbarui catatan, dan DELETE untuk menghapus catatan. Semua metode HTTP dapat digunakan dalam panggilan API. REST API yang dirancang dengan baik mirip dengan situs web yang berjalan di peramban web dengan kemampuan HTTP terintegrasi.

Berdasarkan pembahasan diatas maka penulis menjabarkan rumusan permasalahan yang akan dibahas sebagai berikut: (1) Bagaimana peningkatan pemahaman pengembang melalui penyediaan struktur kode yang jelas? (2) Bagaimana fleksibilitas sistem untuk mengakomodasi perkembangan proses bisnis di masa depan? (3) Dan bagaimana memastikan skalabilitas sistem untuk menangani peningkatan permintaan pengguna?

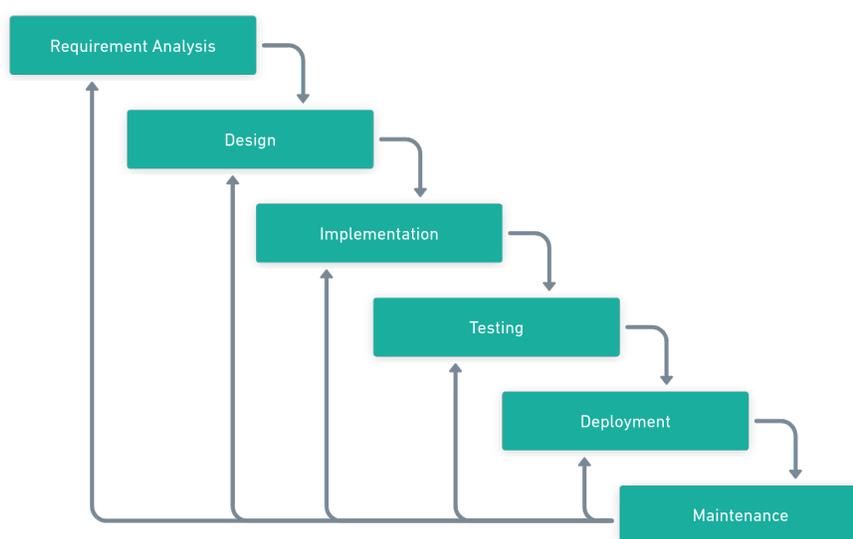
Karena fokus utama penelitian adalah penerapan konsep sistem informasi berbasis API dengan gaya arsitektur REST, peneliti akan membatasi penerapan tersebut pada satu aspek pengajuan surat keterangan dan pengaduan warga, bukan seluruhnya. Peneliti akan membatasi penerapan konsep sistem informasi berbasis API dan gaya arsitektur REST hanya pada bagian pengajuan surat keterangan dan pengaduan warga. Dalam konteks penelitian ini, pengajuan surat keterangan dan pengaduan warga merujuk pada hal-hal berikut: (1) Pengajuan surat keterangan (2) Manajemen dan penandatanganan digital (*digital signature*) surat keterangan (3) Pengajuan dan manajemen laporan pengaduan warga.

Penelitian terkait pengembangan REST API telah dilakukan oleh banyak peneliti. Seperti “Pengembangan REST API Untuk Informasi Pasar Tradisional di Kota Yogyakarta Dengan Metode Incremental” oleh Filiana *et al*, spesifiknya mereka melakukan perancangan sebuah REST API yang menyediakan informasi pasar tradisional di Kota Yogyakarta. Informasi yang ditawarkan termasuk kelas, nama alternatif, lokasi, deskripsi, sejarah, fakta, fasilitas, kekhasan, foto, dan kios [6]. “Sistem

Informasi Pelayanan Online di Mapolresta Bandung” oleh Mubarak et al, berhasil mengembangkan aplikasi, pembangunan infrastruktur dan pelatihan pengguna di di Mapolresta Bandung [7]. Penelitian-penelitian tersebut menjadi salah satu referensi penulis dalam proses pengembangan sistem yang nanti akan diterapkan.

## 2. METODE PENELITIAN

Dalam penelitian ini, proses perancangan REST API dilakukan menggunakan metode waterfall. Alasan penulis menggunakan metode ini karena bersifat sekuensial dan *linear* dimana tahapan pengembangan perangkat lunak dilakukan secara berturut-turut dan setiap tahap harus selesai sebelum memulai tahap berikutnya [8]. Gambar 1 menunjukkan bagaimana tahapan-tahapan waterfall yang dilakukan pada penelitian ini.



Gambar 1. Tahapan Waterfall

### 1. Requirement Analysis (Analisis Kebutuhan)

Penelitian ini menggunakan metode analisis komprehensif untuk memahami kebutuhan pengguna, pemangku kepentingan, dan bisnis yang akan diakomodasi oleh perangkat lunak yang dikembangkan. Proses identifikasi pemangku kepentingan membantu mengenali peran dan pengaruh dari perangkat desa, pengguna akhir, dan tim pengembang. Komunikasi yang efektif, seperti wawancara dan survei, memungkinkan pemahaman mendalam terhadap perspektif, kebutuhan, dan ekspektasi mereka terhadap perangkat lunak. Analisis dokumen dan data yang ada merujuk pada pengumpulan dan analisis informasi terkait sistem backend yang akan dikembangkan, sementara pemahaman tentang teknologi, proses bisnis, dan kebutuhan fungsional dan non-fungsional dijabarkan secara rinci.

Dokumen kebutuhan dibuat dengan cermat untuk memastikan kejelasan, struktur, dan pemahaman yang konsisten oleh semua pemangku kepentingan. Validasi dan verifikasi kebutuhan diimplementasikan untuk memastikan akurasi, relevansi, dan konsistensi kebutuhan yang teridentifikasi, termasuk pemahaman terhadap kemungkinan perubahan selama siklus pengembangan. Dengan pendekatan terstruktur dan komprehensif ini, penelitian ini berhasil

menyusun dasar yang kokoh untuk pengembangan perangkat lunak yang responsif terhadap kebutuhan dan harapan yang teridentifikasi [9].

## **2. Design (Desain)**

Menyusun rancangan arsitektur backend yang terstruktur, penelitian ini merujuk pada hasil analisis kebutuhan pada tahap awal. Desain yang dihasilkan mencakup use case diagram, flowchart, dan struktur database. Pendekatan ini memberikan dasar yang kokoh untuk pengembangan perangkat lunak yang responsif terhadap kebutuhan yang telah diidentifikasi [10].

## **3. Implementation (Implementasi)**

Berdasarkan desain arsitektur backend yang telah disusun, tahap selanjutnya melibatkan pembuatan kode sumber perangkat lunak sesuai dengan rancangan tersebut. Tim pengembang memulai tahap ini dengan menulis kode sumber yang mencakup logika bisnis, manajemen basis data, API, dan komponen-komponen lainnya sesuai dengan arsitektur yang telah dirancang sebelumnya. Selain itu, penggunaan framework pengembangan dapat mempercepat proses implementasi dan memastikan kepatuhan terhadap praktik terbaik. Selama implementasi, pengelolaan database dilakukan sesuai dengan desain struktur database yang telah dirancang, termasuk pembuatan tabel, relasi, dan indeks yang sesuai.

## **4. Testing (Pengujian)**

Dalam tahap ini, penulis menggunakan metode black box testing, dengan Postman sebagai alat pengujian awal sebelum proses deployment. Proses pengujian mencakup tes unit, di mana setiap komponen perangkat lunak diuji terpisah untuk memastikan kinerjanya. Kemudian, pengujian integrasi dilakukan untuk memastikan interaksi yang baik antar komponen [11].

Pengujian fungsional melibatkan pemeriksaan keseluruhan sistem untuk memastikan pemenuhan kebutuhan dan spesifikasi dari tahap analisis kebutuhan. Sementara pengujian kinerja menilai bagaimana sistem berkinerja di bawah beban tertentu, mencakup aspek skalabilitas, kecepatan respon, dan penggunaan sumber daya. Terakhir, pengujian keamanan dilakukan untuk mengidentifikasi dan mengatasi potensi kerentanan atau celah keamanan yang dapat dieksploitasi oleh pihak jahat.

## **5. Deployment (Deploy)**

Dalam tahap ini, penulis melaksanakan proses deployment sehingga API backend yang telah dibuat menjadi tersedia untuk digunakan oleh mobile developer. Proses deployment pada penelitian ini dibatasi sampai dengan upload source code menggunakan version control git, dikarenakan pembagian tugas dalam tim pengembang. Disini peneliti menggunakan platform GitHub sebagai repository.

## **6. Maintenance (Pemeliharaan)**

Tahap akhir dalam pemeliharaan backend melibatkan penggunaan Quick-Fix Model, yang bertujuan untuk mendeteksi kesalahan atau bug dalam sistem dan memastikan performa optimal. Model ini dimulai dengan melakukan perubahan pada kode atau skrip program, diikuti dengan penyesuaian dokumentasi program sesuai kebutuhan. Pendekatan ini dikenal sebagai Model Penyelesaian Cepat [12].

### 3. HASIL DAN PEMBAHASAN

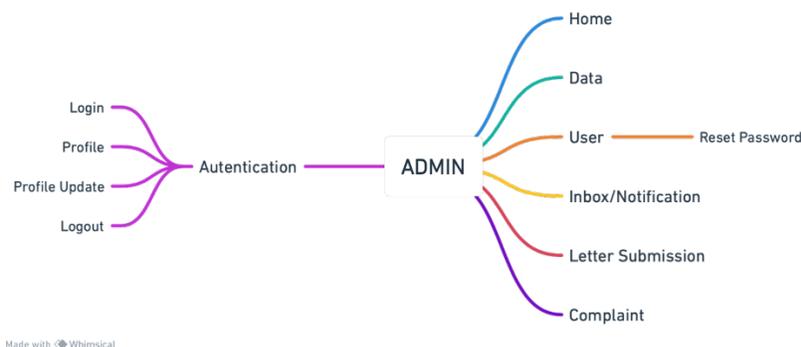
#### 3.1. Requirement Analysis (Analisis Kebutuhan)

##### 3.1.1. Identifikasi Pengguna

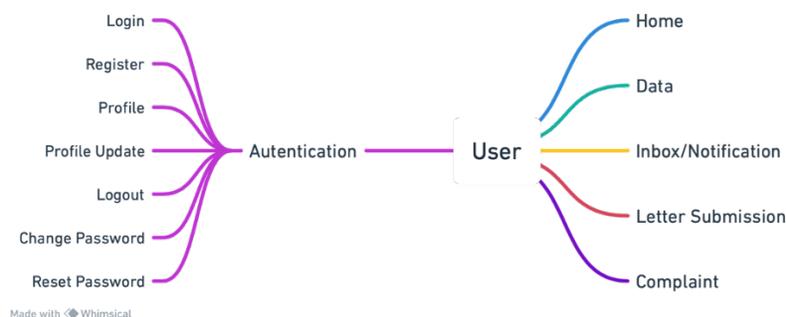
Pada sistem yang akan dikembangkan dalam penelitian ini, pengguna dibagi menjadi dua kategori utama, yaitu Admin dan User. Admin terdiri dari dua jenis, yakni Admin Staff yang bertanggung jawab mengelola proses pengajuan surat dan pengaduan warga di perangkat desa, dan Admin Penandatanganan yang memiliki kewenangan untuk menandatangani surat keterangan yang telah divalidasi oleh Admin Staff. Dalam konteks Desa Tembok, Kepala Desa menjadi peran Admin Penandatanganan. Sementara itu, kategori pengguna lainnya adalah User, yang merujuk pada warga Desa Tembok yang telah terdaftar dalam sistem.

##### 3.1.2. Identifikasi Fitur

Berdasarkan rumusan permasalahan pada bab sebelumnya mengenai kebutuhan dan ruang lingkup penelitian di desa Tembok. Bahwa desa Tembok membutuhkan sistem pelayanan pengajuan surat dan pengaduan warga. Maka dari itu penulis memberikan solusi fitur yang akan dikembangkan pada penelitian ini. Pada Gambar 2 dan Gambar 3 diuraikan sebagai berikut:



Gambar 2. Detail Fitur Admin



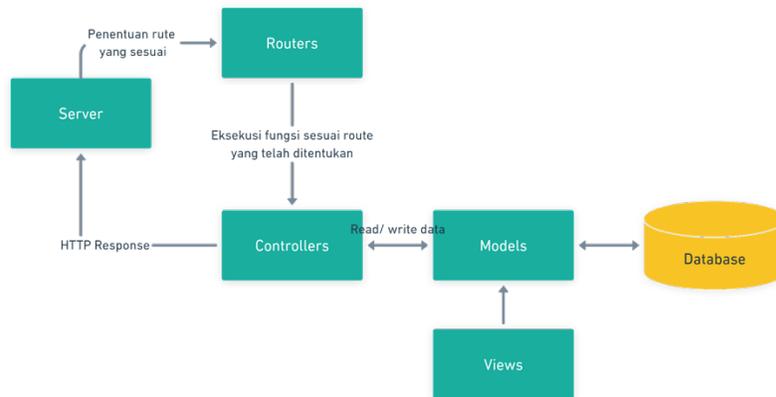
Gambar 3. Detail Fitur User

#### 3.2. Design (Desain)

Setelah kebutuhan dikumpulkan, penulis melanjutkan ke tahap perancangan desain. Pada tahap ini, merancang keseluruhan arsitektur sistem dan memutuskan bagaimana komponen sistem akan berinteraksi dan berkomunikasi satu sama lain. Di tahap ini juga rancangan detail komponen perangkat lunak dibuat. Ini termasuk use case diagram, dan detail teknis lainnya.

### 3.2.1. Rancangan Arsitektur Sistem

Dalam perancangan backend, elemen-elemen kunci seperti database, models, controller, routes, dan server bekerja sama untuk memastikan kinerja optimal. Pada tahap awal, server berfungsi sebagai penghubung antara klien dan aplikasi, menangani permintaan, dan menentukan jalur (routes) berdasarkan jenis permintaan. Melalui pengontrol, fungsi-fungsi terkait jalur dieksekusi, berinteraksi dengan basis data melalui model untuk mengambil dan memproses data, dan hasilnya dikirimkan sebagai respons kepada klien. Gambar 4 merepresentasikan struktur keseluruhan sistem.



Gambar 4. Rancangan Arsitektur Sistem

### 3.2.2. Use Case Diagram

Setelah mempertimbangkan berbagai kemungkinan dalam pengembangan sistem ini, penulis berhasil membuat diagram use case untuk memvisualisasikan persyaratan fungsional dari sistem yang akan dikembangkan. Dengan menunjukkan bagaimana berbagai aktor berinteraksi dengan sistem melalui berbagai use case. Berikut diagram use case yang telah dibuat:



Gambar 5. Use Case Diagram

### 3.3. Implementation (Implementasi)

Berdasarkan rancangan desain yang dibuat, penulis mencoba mengimplementasikan pada penulisan kode sumber berdasarkan rancangan yang telah dibuat. Proses ini melibatkan pemrograman, teknologi yang dipakai, pembuatan komponen perangkat lunak, dan pengembangan fitur-fitur yang dibutuhkan.

Pengembangan REST API tentunya memerlukan spesifikasi teknologi yang mumpuni, mulai dari alat sampai dengan framework yang digunakan. Pada Tabel 1 memberikan informasi terkait spesifikasi teknologi yang digunakan pada penelitian ini.

Tabel 1. Spesifikasi Teknologi

No.	Kriteria	Spesifikasi Teknologi
1	Bahasa Pemrograman	PHP 8.1
2	Framework	Laravel 9
3	API Tester	Postman
4	Database	PostgreSQL 15
5	Version Control System	Git 2.39.2
6	Dependency Manager	Composer 2.5.4
7	Web Services	Apache 2.4.57 (Unix)
8	Code Editor	Visual Studio Code

### 3.4. Testing (Pengujian)

Pada tahap pengujian endpoint penulis menggunakan metode blackbox testing dengan beberapa tahap mulai dari unit testing sampai dengan kesesuaian bisnis proses pada fitur yang telah dikembangkan. Penulis melakukan pengujian dengan dua tahap untuk role user (warga) dan role admin (staf dan penandatangan).

#### 3.4.1. Role User

Pengujian untuk peran user fokus pada evaluasi endpoint dari perspektif warga, menguji bagaimana proses bisnis dalam sistem berjalan sesuai dengan desain yang telah ditetapkan. Langkah ini memberikan pemahaman mendalam mengenai bagaimana pengguna akhir, atau dalam konteks ini, warga, berinteraksi dengan sistem dan sejauh mana sistem mendukung kelancaran proses bisnis yang terlibat.

Hasil pengujian endpoint pada role user dengan request method yang sudah didefinisikan sebelumnya menunjukkan bahwa fitur-fitur yang terkait dengan peran pengguna telah diuji dan berfungsi sesuai dengan harapan.

#### 3.4.2. Role Admin

Sejalan dengan pengujian pada role user, pengujian terhadap role admin juga melibatkan tahap pendefinisian endpoint yang akan diuji. Terdapat 28 endpoint yang perlu dievaluasi, masing-masing dengan request method yang berbeda. Proses ini bertujuan untuk memastikan bahwa fungsi dan fitur yang terkait dengan peran admin dapat beroperasi secara optimal sesuai dengan pengaturan dan kebutuhan yang telah ditetapkan sebelumnya.

Hasil dari pengujian endpoint untuk peran admin, termasuk request method yang telah ditentukan sebelumnya, menunjukkan bahwa fungsi-fungsi yang terkait dengan peran pengguna telah diuji dan beroperasi sesuai dengan yang diinginkan.

### 3.5. *Deployment* (Deploy)

Pada tahap deployment penulis membatasi sampai dengan proses upload source code pada version control GitHub menggunakan standar yang telah disepakati oleh tim pengembang. Selain deployment source code, proses dokumentasi API penulis menggunakan fitur dari Postman yaitu Publish Docs sebagai panduan yang menjelaskan cara menggunakan API, termasuk endpoint, parameter, request, dan response yang didukung oleh API tersebut. Ini membantu pengembang mobile dan web memahami cara berinteraksi dengan API yang telah dikembangkan.

### 3.6. *Maintenance* (Pemeliharaan)

Pada tahap pemeliharaan sistem, koordinasi masalah atau bug dilakukan melalui metode Quick-Fix Model, yang melibatkan beberapa langkah. Pertama, masalah diidentifikasi melalui pelaporan pengguna akhir atau pemantauan sistem. Setelah itu, dilakukan analisis masalah untuk memahami penyebab dan mencari solusi yang sesuai. Perbaikan cepat (quick-fix) diterapkan tanpa mengubah struktur sistem secara signifikan setelah penyebab masalah diketahui. Perbaikan diuji secara fungsional untuk memastikan pemulihan fungsi perangkat lunak. Jika perbaikan lulus uji, diterapkan ke lingkungan produksi atau sistem yang terpengaruh. Sistem dipantau dengan ketat untuk memastikan masalah tidak muncul kembali dan perubahan tidak menimbulkan masalah baru setelah implementasi perbaikan.

## 4. KESIMPULAN

Perancangan Application Programming Interface (API) dengan gaya arsitektur Representational State Transfer (REST) menjadi langkah krusial dalam pengembangan Sistem Pemerintah Berbasis Elektronik (SPBE) di Desa Tembok. API ini, melalui implementasi REST, memberikan kerangka kerja efisien untuk berkomunikasi antara mobile dan web application. Hal ini memudahkan pengembang untuk mengakses dan berinteraksi dengan data serta layanan dalam sistem, menciptakan dasar yang kuat untuk integrasi teknologi informasi dalam penyelenggaraan layanan pemerintah di tingkat desa. Implementasi API REST ini berperan sentral dalam meningkatkan aksesibilitas dan efisiensi SPBE Desa Tembok, memberikan manfaat signifikan bagi masyarakat setempat, dan membuka peluang perkembangan teknologi serta inovasi di masa depan.

## DAFTAR PUSTAKA

- [1] R. Setiawan, D. P. Kartikasari dan B. Rahayudi, "Implementasi Arsitektur Web Server Cluster Menggunakan Single Board Computer untuk Menunjang Kebutuhan High Availability System," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 2, pp. 329-332, 2021.
- [2] A. mii.co.id, "Konsep Restful API Programming (Bagian-1)," Mii.Co.Id, 2021. [Online]. Available: [www.mii.co.id/en/insight/listing/2021/06/21/03/58/konsep-restful-api-programming-bagian-1#](http://www.mii.co.id/en/insight/listing/2021/06/21/03/58/konsep-restful-api-programming-bagian-1#).
- [3] U. Amri, A. dan H. As'ari, "Kebijakan Inovasi dalam Penyelenggaraan Sistem Pemerintahan Berbasis Elektronik (SPBE) di Kota Dumai," *Jurnal Pendidikan Tambusai*, vol. 6, no. 2, 2022.
- [4] R. Fikri, M. F. Amrillah dan H. Selwa, "Kapasitas Adaptif Lokal Pemerintah Desa Dalam Penerapan Sistem Pemerintahan Berbasis Elektronik di Kabupaten Bengkalis," *INTECOMS: Journal of Information Technology and Computer Science*, vol. 2, no. 3, 2020.
- [5] IBM Cloud Education, "IBM.com," 2021. [Online]. Available: [www.redhat.com/en/topics/api/what-is-a-rest-api https://www.ibm.com/cloud/learn/rest-api](https://www.ibm.com/cloud/learn/rest-api). [Diakses 2 Januari 2024].

- [6] A. Filiana, M. N. A. Rini, A. G. Prabawati dan R. A. Samat, “Pengembangan Rest Api Untuk Informasi Pasar Tradisional Di Kota Yogyakarta Dengan Metode Incremental,” *SINTECH (Science and Information Technology) Journal*, vol. 5, no. 1, 2022.
- [7] A. Mubarak, R. Sanjaya, R. T. Prasetyo dan Y. Ramdhani, “Sistem Informasi Pelayanan Online di Mapolresta Bandung,” *Jurnal Abdimas BSI*, vol. 1, no. 1, pp. 1-6, 2018.
- [8] H. Ms, “Metode Waterfall : Pengertian, Tahapan, Kekurangan dan Keunggulan,” *Seriksa.com*, vol. 3, 2021.
- [9] A. A. Kadim, . I. K. Sutriana dan . I. H. Masir, “Perancangan Sistem Aplikasi Layanan Kelurahan Berbasis Web,” *Jambura Journal of Informatics*, vol. 4, no. 1, pp. 38-48, 2022.
- [10] S. M. N. Muhammad, F. A. Mauladi, R. Kurniawan dan R. Sanjaya, “Pengembangan Sistem Informasi Kawasan Agrowisata Menggunakan Konsep Model View Control berbasis Web,” *Edumatic: Jurnal Pendidikan Informatika*, vol. 6, no. 1, pp. 88-97, 2022.
- [11] Uminingsih, M. Nur Ichsanudin, . M. Yusuf dan S. S. Suraya, “Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula,” *STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer*, vol. 1, no. 2, pp. 1-8, 2022.
- [12] J. D. SUSATYONO, “Pemeliharaan Perangkat Lunak (Kategori, Model Pemeliharaan Dan Proses Pemeliharaan),” 2021.