

PEMBUATAN *WEB SERVICE* PADA APLIKASI SIJAB DENGAN METODE REST

Denis Akbar, Freza Riana, Fitriah Satrya

Teknik Informatika, Universitas Ibn Khaldun Bogor
Jalan Sholeh Iskandar Kedung badak Tanah Sareal Kota Bogr , Indonesia
bzndenis@gmail.com

ABSTRAK

Penelitian ini bertujuan untuk membuat sistem pemantauan lalu lintas berbasis *web service* menggunakan metode REST pada aplikasi Sistem Informasi Jalan Raya Kota Bogor (SIJAB). Latar belakang penelitian ini didasari oleh peningkatan jumlah kendaraan dan kompleksitas lalu lintas yang semakin tinggi, sehingga diperlukan solusi yang efektif untuk memantau dan mengelola kondisi lalu lintas. Penggunaan *web service* dengan metode REST diterapkan untuk menyediakan layanan yang akurat dan *real-time* bagi pengguna jalan. Penelitian ini menggunakan pendekatan *waterfall* dalam pengembangan, dengan langkah-langkah yang terstruktur. Lumen Laravel digunakan sebagai *framework backend* API untuk membangun aplikasi *web API* yang ringan dan cepat. Sistem Informasi Jalan Raya Kota Bogor (SIJAB) dirancang sebagai *platform* terintegrasi yang meliputi peta kemacetan, CCTV lalu lintas, dan jalur trayek angkutan umum. Data lalu lintas disajikan melalui aplikasi *mobile* dan *website* untuk memberikan informasi yang berguna kepada pengguna jalan. Hasil penelitian ini menunjukkan bahwa sistem yang dibangun berhasil meningkatkan efisiensi pendataan lalu lintas dan menyediakan informasi yang akurat serta *real-time* kepada pengguna. Selain itu, sistem ini juga mampu menciptakan pengendalian lalu lintas yang lebih efektif di Kota Bogor. Manfaat dari penelitian ini mencakup peningkatan keselamatan lalu lintas, kenyamanan pengguna jalan, serta mendorong penggunaan teknologi *web service* dalam pengembangan aplikasi terkait pemantauan lalu lintas.

Kata kunci: *Web service, REST, Sistem Informasi Jalan Raya, Lumen Laravel, SIJAB.*

1. PENDAHULUAN

Jalan memiliki peran penting dalam memfasilitasi pergerakan masyarakat, sesuai dengan ketentuan yang diatur dalam Undang-Undang No. 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan [1]. Namun, seiring dengan peningkatan jumlah kendaraan, terutama di kota-kota padat penduduk, timbul masalah lalu lintas yang serius, seperti kemacetan [2][3]. Dengan bertambahnya populasi dan jumlah kendaraan, penting untuk memiliki sistem pemantauan lalu lintas yang efektif [4], yang dapat menyediakan informasi akurat dan *real-time* [5].

Salah satu cara untuk menghubungkan sistem informasi *website* dan perangkat *mobile* adalah dengan menggunakan teknologi *web service* yang berfungsi sebagai jembatan komunikasi antar aplikasi [6]. Metode yang sering digunakan dalam implementasi layanan *web* adalah REST, yang memungkinkan sistem berkomunikasi melalui protokol HTTP dengan metode standar seperti *GET*, *POST*, *PUT*, dan *DELETE* [7][8][9].

Penelitian terkait *web service* menunjukkan keberhasilan dalam memfasilitasi akses dan fleksibilitas pengguna, seperti yang dilakukan oleh Reza dan timnya untuk sistem penelitian dan pengabdian masyarakat [10], serta oleh Diki dan timnya dalam sinkronisasi transaksi data aplikasi fasilitas keuangan [11].

Sistem Informasi Jalan Raya Kota Bogor (SIJAB) adalah aplikasi yang dirancang untuk memberikan informasi kepada pengguna jalan tentang kondisi lalu lintas di Kota Bogor [12]. SIJAB mencakup peta kemacetan, CCTV lalu lintas, dan jalur

trayek angkutan umum, serta menampilkan data lalu lintas melalui aplikasi *mobile* dan *website* [13].

Penelitian ini berfokus pada pembuatan *web service* dengan metode REST untuk SIJAB, yang diharapkan dapat meningkatkan efektivitas dan efisiensi pendataan lalu lintas di Kota Bogor, serta memberikan informasi lalu lintas yang dinamis dan *real-time*. Pengembangan aplikasi *web service* berbasis REST untuk SIJAB bertujuan untuk menciptakan sistem pengendalian lalu lintas yang lebih efektif, meningkatkan keselamatan lalu lintas, dan memberikan kenyamanan serta kepuasan kepada pengguna jalan di Kota Bogor.

2. TINJAUAN PUSTAKA

2.1. Lumen Laravel

Lumen Laravel adalah salah satu *framework* PHP yang dikembangkan oleh Taylor Otwell sebagai versi ringan dari Laravel. *Framework* ini dirancang khusus untuk membangun aplikasi *web API* yang cepat dan efisien. Lumen Laravel dilengkapi dengan berbagai fitur yang mempermudah proses pengembangan aplikasi *web*, seperti sistem routing, *database migration*, *middleware*, dan lain-lain. Kelebihan utama Lumen Laravel adalah kemampuannya mengoptimalkan kinerja aplikasi *web* sehingga dapat menangani banyak permintaan dari klien secara bersamaan [14].

2.2. *Application Programming Interface* (API)

Application Programming Interface (API) adalah antarmuka yang memungkinkan dua atau lebih aplikasi untuk berkomunikasi dan bertukar data. API

memungkinkan program atau sistem berinteraksi dengan komponen lain secara standar, sehingga mempermudah integrasi dan pengembangan aplikasi. API sering digunakan dalam pengembangan aplikasi *web* dan *mobile* [15].

2.3. Representational State Transfer (REST)

Representational State Transfer (REST) adalah gaya arsitektur yang digunakan untuk membangun *web service*. REST memanfaatkan prinsip-prinsip arsitektur *web* untuk menciptakan *web service* yang mudah digunakan dan skalabel. REST dapat menjadi solusi efektif dalam membangun *web service* yang mudah diakses oleh berbagai jenis klien dan membantu meningkatkan performa *web service* [15].

Tabel 1. Operasi CRUD

Operasi	Metode	Proses
CREATE	POST/PUT	Digunakan untuk membuat atau menambah data
READ	GET	Digunakan untuk menampilkan data
UPDATE	PUT	Digunakan untuk mengubah atau memperbaiki data
DELETE	DELETE	Digunakan untuk menghapus data

2.4. Web service

Web service adalah layanan yang disediakan oleh suatu sistem atau aplikasi melalui jaringan, memungkinkan akses oleh sistem lain melalui protokol standar seperti HTTP. *Web service* memungkinkan interoperabilitas antar aplikasi yang berbeda dan mendukung integrasi sistem yang terdistribusi [15].

2.5. MySQL

MySQL adalah salah satu sistem manajemen basis data (DBMS) yang paling populer dan banyak digunakan di dunia. DBMS ini merupakan perangkat lunak sumber terbuka (*open-source*) yang dikembangkan oleh Oracle Corporation. MySQL menyediakan lingkungan yang kuat dan efisien untuk mengelola, menyimpan, dan mengakses data secara efisien. MySQL menggunakan bahasa SQL (*Structured Query Language*) untuk mengelola basis data. Dengan SQL, pengguna dapat dengan mudah melakukan berbagai operasi seperti pembuatan, penghapusan, pembaruan, dan pengambilan data. MySQL juga mendukung fitur-fitur canggih seperti indeks, transaksi, pemantauan kinerja, dan replikasi, yang memungkinkan pengguna untuk mengoptimalkan kinerja basis data dan memastikan ketersediaan data secara maksimal [16].

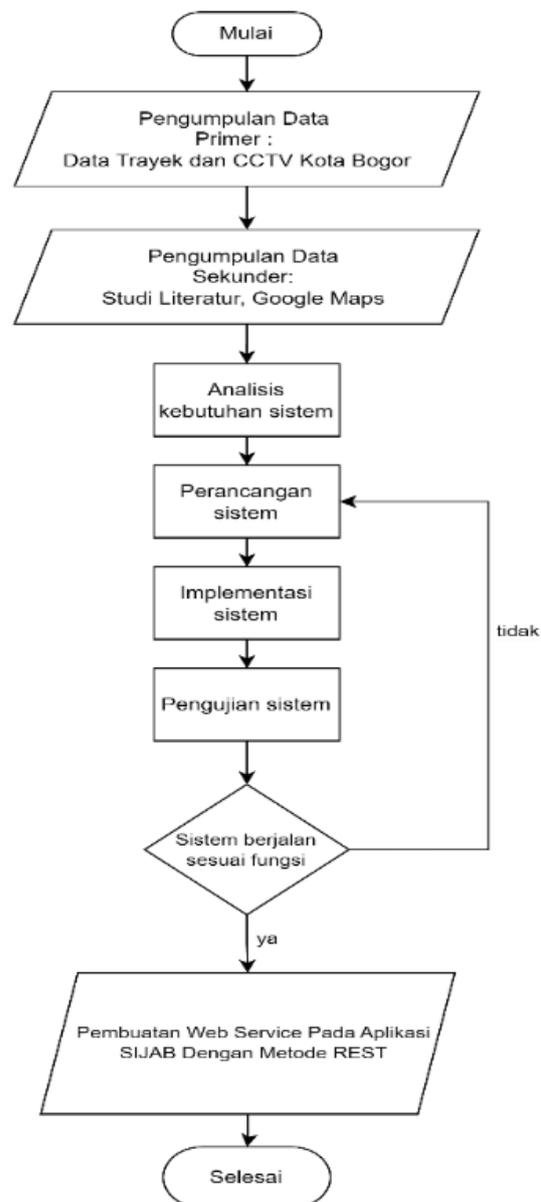
2.6. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa visual yang digunakan untuk pemodelan, spesifikasi, pembangunan, dan dokumentasi sistem perangkat lunak. UML telah menjadi landasan penting dalam pengembangan perangkat lunak karena

kemampuannya menggambarkan berbagai aspek sistem secara jelas dan terstruktur. Dalam pengembangan perangkat lunak, UML bukan hanya alat atau notasi semata, tetapi juga kerangka kerja yang luas yang mencakup beragam elemen dan konsep untuk menggambarkan sistem dengan lebih baik [17].

3. METODE PENELITIAN

Metode *waterfall* adalah suatu pendekatan linear dan berurutan dalam pengembangan proyek yang terdiri dari serangkaian tahapan yang saling terkait. Pendekatan ini sering digunakan dalam pengembangan perangkat lunak dan proyek-proyek teknologi informasi lainnya. Dalam konteks penelitian ini, pendekatan *waterfall* digunakan untuk mengarahkan proses penelitian dengan langkah-langkah yang terstruktur dan terencana. Metode penelitian ditunjukkan pada Gambar 1.



Gambar 1. Metode Penelitian *Waterfall*

3.1. Pengumpulan Data

Tahap ini adalah langkah penting dalam perjalanan penelitian, di mana dilakukan serangkaian kegiatan yang bertujuan untuk mengumpulkan dan menyusun data guna memfasilitasi proses analisis yang mendalam terkait topik penelitian. Dalam konteks ini, penulis akan membahas secara rinci mengenai proses pencarian dan pengumpulan data yang meliputi dua bagian utama, yaitu:

a. Data Primer

Data primer adalah jenis data yang diperoleh secara langsung dari sumber pertama atau melalui pengamatan langsung di lapangan. Jenis data ini memiliki keunggulan dalam hal akurasi dan relevansi karena diperoleh langsung dari sumber yang terlibat dalam situasi yang diteliti. Dalam penelitian ini, data primer diperoleh melalui proses wawancara dengan salah satu ketua divisi dari pihak Dinas Perhubungan Kota Bogor.

b. Data Sekunder

Data sekunder adalah jenis data yang diperoleh dari sumber-sumber lain di luar peneliti, seperti jurnal ilmiah, buku elektronik (e-book), situs *web*, dan dokumen-dokumen resmi pemerintah. Metode perolehan data ini dilakukan melalui studi literatur yang berkaitan dengan aspek-aspek yang relevan dengan penulisan tentang layanan *web service*.

3.2. Analisis

Pada tahap ini, dilakukan proses analisis untuk menentukan kebutuhan sistem dan menganalisis pengolahan data yang telah diperoleh. Analisis ini mencakup beberapa aspek penting, seperti analisis kebutuhan fungsional sistem, analisis kebutuhan non-fungsional, analisis arsitektur sistem, evaluasi sistem yang sedang berjalan, dan perencanaan sistem yang akan dibangun. Dalam analisis kebutuhan fungsional sistem, dilakukan identifikasi dan deskripsi rinci terhadap fitur-fitur dan fungsionalitas yang harus ada dalam sistem yang akan diimplementasikan.

3.3. Perancangan

Pada tahap ini, sistem dirancang menggunakan Unified Modeling Language (UML) sebagai alat bantu. Dalam proses perancangan ini, berbagai diagram UML seperti diagram *use case*, diagram class, diagram komponen, dan diagram *deployment* digunakan untuk menggambarkan struktur dan interaksi antar komponen dalam sistem secara lebih terstruktur dan visual. Selanjutnya, desain *database* yang didapatkan dari tahap analisis akan diterjemahkan menjadi desain *database* yang lebih terperinci. Selain itu, perancangan pemasangan *web service* menggunakan metode *Representational State Transfer* (REST) juga dilakukan, yang melibatkan penentuan endpoint, format data yang digunakan, serta cara interaksi antara berbagai komponen melalui *web service*.

3.4. Implementasi

Dalam tahap implementasi ini, desain sistem diubah menjadi bentuk yang dapat dipahami dan dieksekusi oleh komputer. Ini dilakukan dengan menggunakan bahasa pemrograman berbasis PHP, dengan menggunakan *framework* Laravel untuk pengembangan aplikasi secara keseluruhan, dan Lumen sebagai *micro framework* yang terintegrasi dengan Laravel.

3.5. Pengujian

Pada tahap ini, fokusnya adalah pada pengujian sistem yang telah dibangun. Tujuannya adalah untuk mengidentifikasi potensi kekurangan dan kesalahan yang mungkin ada dalam sistem. Metode yang digunakan adalah metode *blackbox*, di mana setiap *web service* yang telah dibuat akan diperiksa secara independen untuk memastikan bahwa fungsi-fungsi yang diharapkan dapat berjalan sesuai rencana. Pengujian dilakukan menggunakan perangkat lunak *postman*, yang memungkinkan pengiriman permintaan ke berbagai *web service* dan pemeriksaan respons yang dihasilkan.

4. HASIL DAN PEMBAHASAN

4.1. Analisis Kebutuhan

Pada bagian ini, dilakukan serangkaian proses analisis terhadap kebutuhan sistem. Proses ini mencakup beberapa aspek penting seperti analisis kebutuhan fungsional sistem, kebutuhan non-fungsional yang harus dipenuhi, struktur arsitektur sistem yang akan dibangun, evaluasi sistem yang sudah berjalan, dan perencanaan untuk sistem yang akan dikembangkan. Setiap elemen ini dianalisis secara cermat untuk memastikan bahwa semua aspek yang relevan telah dipertimbangkan secara menyeluruh.

4.2. Analisis Kebutuhan Fungsional

Tingkat kebutuhan fungsional mengacu pada semua kebutuhan sistem yang menggambarkan berbagai layanan di dalam *web service* yang diperlukan untuk memenuhi kebutuhan aplikasi pencarian fasilitas keuangan yang berbasis *mobile* Android dan *web*. Sesuai dengan kebutuhan tersebut, modul-modul yang akan diimplementasikan melalui *web service* dapat dikelompokkan sebagai berikut:

a. Modul Login Admin

Modul ini digunakan oleh admin untuk login ke halaman dashboard.

b. Modul CCTV

Modul ini digunakan untuk mendapatkan data, menyimpan, mengubah dan menghapus CCTV.

c. Modul Fasum

Modul ini digunakan untuk mendapatkan data, menyimpan, mengubah dan menghapus fasilitas umum seperti terminal, stasiun, rest area.

- d. Modul Berita
Modul ini digunakan untuk mendapatkan data, menyimpan, mengubah dan menghapus berita seputar jalan raya Kota bogor.
- e. Modul Polantas
Modul ini digunakan untuk mendapatkan data, menyimpan, mengubah dan menghapus Polres, Polsek dan pos polisi.
- f. Modul Trayek
Modul ini digunakan untuk mendapatkan data, menyimpan, mengubah dan menghapus trayek angkot dan bis kota.
- g. Modul User
Modul ini digunakan untuk mendapatkan data username dan nama untuk admin.
- h. Modul Publik
Modul ini digunakan untuk mendapatkan data dari semua endpoint dengan metode *GET* dengan endpoint terbuka untuk umum.

4.3. Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional bertujuan utama untuk mengidentifikasi aspek-aspek yang berinteraksi secara mendalam dengan sistem yang sedang berjalan. Dalam upaya mencapai pemahaman yang lebih menyeluruh, ada beberapa kebutuhan non-fungsional yang memerlukan perhatian khusus dalam konteks sistem ini. Berikut adalah beberapa contoh kebutuhan non-fungsional yang menjadi fokus analisis:

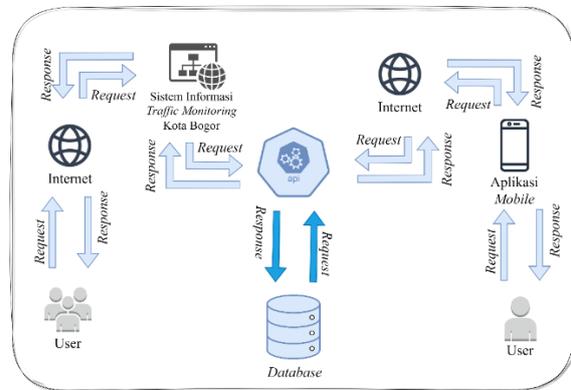
- a. Ketepatan (*Correctness*)
Web service menyajikan data – data yang benar sesuai dengan data yang dibutuhkan untuk aplikasi sistem informasi jalan raya Kota Bogor.
- b. Interoperabilitas (*Interoperability*)
Web service dapat digunakan oleh platform dan sistem operasi yang berbeda.
- c. Skalabilitas (*Scalability*)
Web service memiliki kemampuan suatu sistem untuk mengatasi peningkatan beban atau volume kerja tanpa mengorbankan performa atau kinerja.
- d. Kemampuan untuk di uji (*Testability*)
Web service dapat diuji coba pada aplikasi *POSTman* sebagai client untuk setiap pengujian endpoint yang telah dibuat.
- e. Portabilitas (*Portability*)
Web service dapat di akses melalui jaringan internet pada setiap endpoint yang telah dibuat.

4.4. Analisis Arsitektur Sistem

Tujuan utama dari *web service* ini adalah memudahkan akses data secara efisien dan andal di berbagai platform, termasuk platform *mobile* dan *web*. Selain itu, perlu diperhatikan bahwa potensi pemanfaatan *web service* ini melibatkan interaksi dengan sistem lain yang juga membutuhkan akses terhadap data yang tersimpan dalam sistem informasi jalan raya Kota Bogor.

Dalam konteks ini, penting untuk mengakui perbedaan karakteristik dan kebutuhan antara aplikasi-

aplikasi yang terhubung dengan *web service*. Aplikasi sistem informasi memiliki peran penting dalam memulai transaksi data dengan mengirimkan permintaan kepada *web service*. Proses komunikasi ini dibantu oleh struktur data yang diadopsi dalam file *web service*. Dalam hal ini, format data yang digunakan adalah *JSON*, yang memungkinkan penyampaian informasi secara efisien dan ringkas. Penggunaan *JSON* sebagai format pertukaran data mempermudah transaksi antara aplikasi sistem informasi jalan raya Kota Bogor dengan *web service*. Ketika aplikasi ini mengirimkan permintaan data, respons yang diterima akan berbentuk file *JSON*. File *JSON* tersebut kemudian akan diolah sesuai dengan logika aplikasi dan digunakan untuk memenuhi kebutuhan dari masing-masing aplikasi. Penjelasan arsitektur sistem informasi jalan raya Kota Bogor dapat dilihat pada Gambar 2.



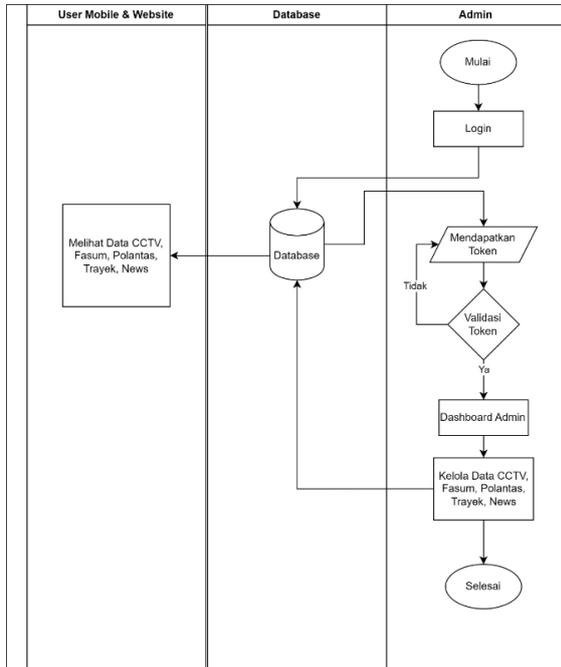
Gambar 2. Analisis Arsitektur Sistem

4.5. Analisis Sistem yang Akan Dibuat

Analisis sistem yang direncanakan untuk penelitian ini akan menjelaskan dengan lebih rinci bagaimana keseluruhan sistem akan dijalankan serta langkah-langkah yang akan dilalui dalam pelaksanaannya. Fokus utama dari rancangan sistem ini adalah menyediakan layanan *web service* yang akan terintegrasi dengan aplikasi sistem informasi jalan raya di Kota Bogor. Dalam upaya mencapai tujuan tersebut, juga dijelaskan serangkaian proses validasi yang akan diterapkan, termasuk proses validasi saat pengguna melakukan login dan penggunaan akses token pada berbagai endpoint yang memiliki izin khusus. Langkah-langkah ini sangat penting untuk menjaga keamanan sistem.

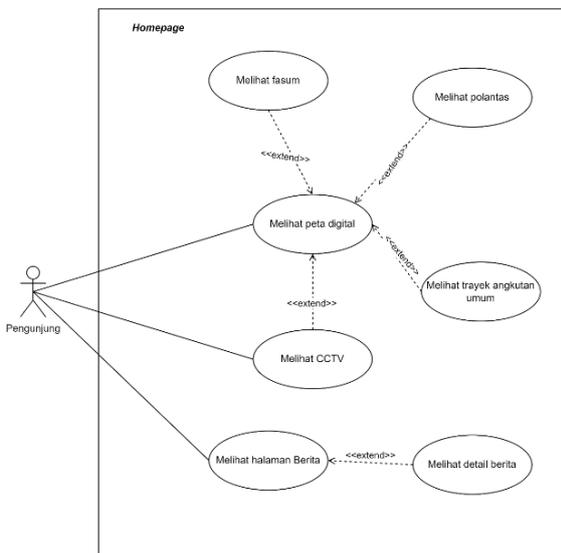
Selain itu, sistem yang dibuat juga memperhatikan aspek keamanan melalui pengembangan dashboard admin yang dilengkapi dengan mekanisme penyimpanan, pembaruan, dan penghapusan data. Tindakan-tindakan pengamanan ini memberikan lapisan perlindungan tambahan terhadap data sensitif yang dikelola dalam sistem. Sistem ini akan memberikan akses kepada pengguna umum, yang dapat mengakses berbagai modul yang telah diterapkan dalam sistem, seperti akses ke feed CCTV, informasi fasilitas umum, berita terkini, rincian

polantas, serta informasi rute trayek. Gambaran analisis sistem yang akan dibuat dapat dilihat pada Gambar 3.



Gambar 3. Analisis Sistem yang Akan Dibuat

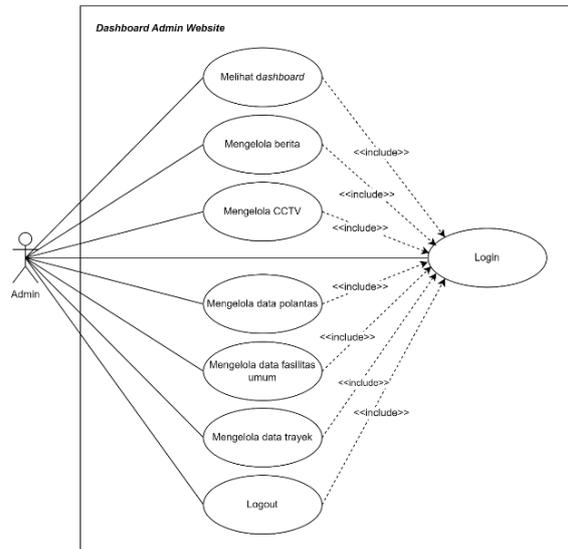
4.6. Perancangan Sistem



Gambar 4. Use case Diagram Homepage

Pada tahap ini, dilakukan penyusunan desain rancangan dengan menggunakan bahasa pemodelan UML untuk menjelaskan struktur sistem melalui diagram use case yang rinci, serta merinci perancangan basis data berdasarkan hasil analisis sebelumnya. Tahap ini juga mencakup desain pemasangan layanan web dengan menerapkan metode REST untuk memastikan komunikasi antarmuka yang efisien dan scalable. Seluruh proses perancangan ini

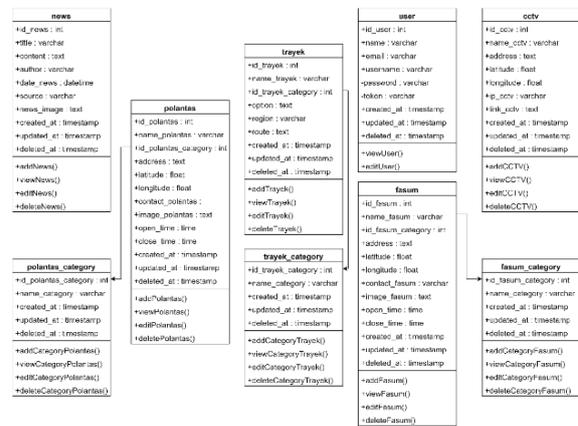
melibatkan analisis mendalam untuk memastikan kesesuaian fungsionalitas sistem yang akan menjadi dasar implementasi selanjutnya yang ditujukan pada gambar 4 dan 5.



Gambar 5. Use case Diagram Dashboard Admin

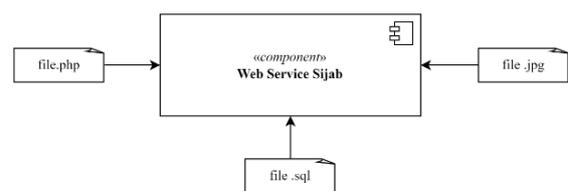
4.7. Class Diagram

Diagram Kelas digunakan untuk menggambarkan hubungan antara kelas, kumpulan kelas, kolaborasi, dan hubungan antara kelas serta antarmuka. Diagram Kelas dapat dilihat pada Gambar 6.



Gambar 6. Class Diagram

4.8. Componen Diagram

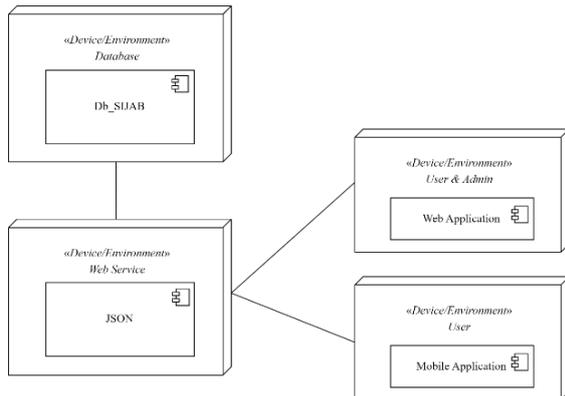


Gambar 7. Componen Diagram

Diagram komponen menggambarkan struktur dan hubungan antara komponen-komponen, serta ketergantungan yang terbentuk antara komponen-komponen tersebut dengan sistem secara keseluruhan. Diagram komponen dapat dilihat pada Gambar 7.

4.9. Deployment Diagram

Diagram *deployment* menggambarkan hubungan antara komponen-komponen perangkat lunak dan perangkat keras, serta menjelaskan konfigurasi setiap komponen saat beroperasi. Diagram penyebaran dapat dilihat pada Gambar 8.



Gambar 8. Deployment Diagram

4.10. Perancangan Database

Basis data yang akan diimplementasikan terdiri dari total 9 tabel dalam strukturnya. Skema lengkap basis data dapat dilihat pada Gambar 9.



Gambar 9. Database

4.11. Hasil Implementasi

Dalam hasil implementasi, terdapat 44 (Empat Puluh Empat) layanan yang telah dikembangkan dalam *web service* ini dengan 8 modul utama, seperti berikut ini:

4.12. Modul Admin

Modul admin adalah layanan yang memungkinkan admin masuk ke dashboard untuk mengelola berbagai fungsi yang akan ditampilkan kepada pengguna *website* dan *mobile*. Dokumentasi lengkap layanan Login admin dapat ditemukan dalam Tabel 2.

Tabel 2. Modul Admin

Modul Admin	
URL	https://api.skripsijoss.my.id/Login
Method	POST
Parameter	username, password
Success Response	<pre> "message": "Login Berhasil", "code": 200, "result": { "token": "0IFxIPFJrI9cIR9pHwUR6lHr8TTHat9d 5QbuSgS34Qc02gJUfIHBc2pssIaAHbip wyZKqbkIbnrdABIJ" } </pre>

4.13. Modul CCTV

Melihat data pada modul CCTV merupakan service yang berfungsi untuk admin melihat data CCTV yang ada pada dashboard admin. Adapun dokumentasi service melihat data CCTV dapat dilihat pada Tabel 3.

Tabel 3. Modul CCTV

Modul CCTV	
URL	https://api.skripsijoss.my.id/cctv
Method	GET
Parameter	token
Success Response	<pre> "status": 200, "success": true, "message": "Berhasil Mendapatkan Data", "total_data": 44, "data": [{ "id": 75, "name_cctv": "Jagorawi - U - Turn Ciawi [B]", "address_cctv": "Jalan Tol Jagorawi, RW 09 Baranangsiang, Bogor Timur, Bogor, West Java, 16143, Indonesia", "lat_cctv": "-6.601688529142781", "lng_cctv": "106.811131238937390", "ip_cctv": "127.0.0.1", "link_cctv": "https://jid.jasamarga.com/vmzm8/nph- zms?mode=jpegdanmonitor=254danscale=100da nmaxfps=30danbuffer=1000danuser=viewdanpas s=liataja", "created_at": "2023-06-26T03:28:43.000000Z", "updated_at": "2023-06-26T03:29:37.000000Z", "deleted_at": null }] </pre>

4.14. Modul Fasum

Menampilkan data pada modul fasum merupakan service yang berfungsi untuk admin melihat data fasum yang ada pada dashboard admin. Adapun dokumentasi service melihat data fasum dapat dilihat pada Tabel 4.

Tabel 4. Data Fasum

Modul Fasum	
URL	https://api.skripsijoss.my.id/fasum
Method	GET
Parameter	token
Success Response	<pre> "status": 200, "success": true, "message": "Berhasil Mendapatkan Data", "total_data": 56, "data": [{ "id": 1, "name_fasum": "Stasiun Bogor", "category_fasum": { "id": 2, "name_category": "Stasiun", "created_at": "2023-06-19T16:19:09.000000Z", "updated_at": "2023-06-19T16:19:09.000000Z", "deleted_at": null } }] </pre>

4.15. Modul Polantas

Melihat data pada modul polantas merupakan service yang berfungsi untuk admin melihat data polantas yang ada pada dashboard admin. Adapun dokumentasi service melihat data polantas dapat dilihat pada Tabel 5.

Tabel 5. Modul Polantas

Modul Polantas	
URL	https://api.skripsijoss.my.id/polantas
Method	GET
Parameter	token
Success Response	<pre> "status": 200, "success": true, "message": "Berhasil Mendapatkan Data", "total_data": 15, "data": [{ "id": 6, "name_polantas": "Polsek Bogor Utara", "category_polantas": { "id": 1, "name_category": "Polsek", "created_at": "2023-06-19T16:19:02.000000Z", "updated_at": "2023-06-19T16:19:02.000000Z", "deleted_at": null } }] </pre>

4.16. Modul Trayek

Menampilkan data pada modul trayek merupakan service yang berfungsi untuk admin melihat data fasum yang ada pada dashboard admin. Adapun dokumentasi service melihat data trayek dapat dilihat pada Tabel 6.

Tabel 6. Modul Trayek

Modul Trayek	
URL	https://api.skripsijoss.my.id/trayek
Method	GET
Parameter	token
Success Response	<pre> ""status": 200, "success": true, "message": "Berhasil Mendapatkan Data", "total_data": 10, "data": [{ "id": 9, "name_trayek": "BisKita K1", "category_trayek": { "id": 2, "name_category": "Biskita", "created_at": "2023-06-19T16:19:09.000000Z", "updated_at": "2023-06-19T16:19:09.000000Z", "deleted_at": null }, "image_trayek": "1691471239.png", "option": [{"options": {}, "latLng": {"lat": -6.5700394873063575, "lng": 106.75485447049142}, "name": "Terminal Bubulak, Jalan Gardu Raya, Bogor Barat, Bogor, West Java, Java, 16112, Indonesia", "_initHooksCalled": true}, {"options": {}, "latLng": {"lat": -6.565030566192793, "lng": 106.81039631366731}], "region": "Bogor Kota", "route": "Terminal Bubulak - Cidangiang", "created_at": "2023-08-08T05:07:19.000000Z", "updated_at": "2023-08-08T05:07:19.000000Z", "deleted_at": null }], </pre>

4.17. Modul Berita

Melihat data pada modul berita merupakan service yang berfungsi untuk admin melihat data berita yang ada pada dashboard admin. Adapun dokumentasi service melihat data berita dapat dilihat pada Tabel 7.

Tabel 7. Modul Berita

Modul Berita	
URL	https://api.skripsijoss.my.id/berita
Method	GET
Parameter	token
Success Response	<pre> "status": 200, "success": true, "message": "Berhasil Mendapatkan Data", </pre>

Modul Berita	
	<pre> "total_data": 12, "data": [{ "id": 35, "title": "Menggali Kecelakaan Jalan Raya Bogor: Penyebab Mengerikan yang Harus Anda Ketahui", "content": "Bogor, Kota yang terkenal dengan keindahan alam dan hawa sejuknya, juga memiliki catatan kelam yang tidak boleh diabaikan—kecelakaan jalan raya. "author": "admin", "date_news": "2023-05-24", "source": "Bogorianssd", "news_image": "1688973842.jpg", "created_at": "2023-05-24T09:07:55.000000Z", "updated_at": "2023-07-10T07:24:02.000000Z", "deleted_at": null }].] </pre>

4.18. Modul Publik

Modul data pelayanan publik merupakan service yang berfungsi sebagai endpoint terbuka untuk melihat total cctv, trayek, halte, pos polisi Adapun dokumentasi service melihat data Count pelayanan publik dapat dilihat pada Tabel Tabel 8.

Tabel 8. Modul Publik

Modul Publik	
URL	https://api.skripsijoss.my.id/open/Count_pelayanan
Method	GET
Parameter	token
Success Response	<pre> "status": 200, "success": true, "message": "Berhasil Mendapatkan Data", </pre>

Modul Publik	
	<pre> "total_data": 4, "data": { "cctv": 44, "trayek": 10, "terminal": 47, "pos": 8 } "deleted_at": null }] </pre>

4.19. Modul User

Modul user merupakan service yang berfungsi untuk admin mengubah data user yang ada pada dashboard admin. Adapun dokumentasi service modul data user dapat dilihat pada Tabel 9.

Tabel 9. Modul User

Modul User	
URL	https://api.skripsijoss.my.id/user
Method	POST
Parameter	token
Success Response	<pre> { "message": "Informasi pengguna berhasil diperbarui", "code": 200 } </pre>

4.20. Pengujian

Pada bagian pengujian ini, tujuannya adalah untuk mengamati hasil eksekusi menggunakan sejumlah data uji dan memeriksa fungsionalitas perangkat lunak yang telah dikembangkan. Metode yang digunakan dalam pengujian pada penelitian ini adalah metode blackbox. Metode blackbox digunakan untuk memvalidasi hasil yang dihasilkan oleh sistem saat menerima suatu perintah. Informasi terkait pengujian endpoint dapat ditemukan dalam Tabel 10.

Tabel 10. Pengujian

No	Nama Modul	Skenario Pengujian	Hasil yang Diharapkan	Hasil Uji
1	Admin	Kirim request dengan method <i>POST</i> beserta data admin	Success response dengan status code 200	Berhasil
2	CCTV	Kirim request dengan method <i>GET, POST, DELETE</i> beserta data CCTV	Success response dengan status code 200	Berhasil
3	Fasum	Kirim request dengan method <i>GET, POST, DELETE</i> beserta data fasum	Success response dengan status code 200	Berhasil
4	Polantas	Kirim request dengan method <i>GET, POST, DELETE</i> beserta data polantas	Success response dengan status code 200	Berhasil
5	Trayek	Kirim request dengan method <i>GET, POST, DELETE</i> beserta data trayek	Success response dengan status code 200	Berhasil
6	Berita	Kirim request dengan method <i>GET, POST, DELETE</i> beserta data berita	Success response dengan status code 200	Berhasil
7	Publik	Kirim request dengan method <i>GET</i> beserta parameter yang dibutuhkan	Success response dengan status code 200	Berhasil
8	User	Kirim request dengan method <i>GET, POST</i> beserta data user	Success response dengan status code 200	Berhasil

Tabel 10 menunjukkan bahwa seluruh skenario pengujian pada berbagai modul telah berhasil. Setiap request yang dikirim, baik menggunakan method *GET, POST*, maupun *DELETE*, menghasilkan respons

yang sesuai dengan yang diharapkan, yaitu status code 200 (*Success*). Hasil ini menegaskan bahwa semua fungsi utama dalam sistem bekerja dengan baik dan memenuhi kebutuhan yang telah ditentukan.

5. KESIMPULAN DAN SARAN

Dari hasil penelitian pembuatan layanan *web service* dengan metode REST, perancangan dan pengujian yang telah dilakukan menghasilkan kesimpulan bahwa pembuatan layanan *web service* pada Aplikasi SIJAB telah berhasil meningkatkan efektivitas operasional dalam menyediakan informasi kepada pengguna. Keberhasilan aplikasi dalam memberikan layanan yang lebih responsif dan terintegrasi melalui berbagai *platform*, termasuk *web apps* dan *mobile apps* berbasis Android, telah terbukti. Hasil pengujian menggunakan metode blackbox menunjukkan bahwa semua modul yang diuji, seperti Admin, CCTV, Fasum, Polantas, Trayek, Berita, Publik, dan User, berhasil memenuhi ekspektasi dengan *respons status code 200 (Success)*. Dengan adanya layanan *web service* di kedua *platform* tersebut, diharapkan aplikasi SIJAB dapat memberikan kemudahan akses dan fleksibilitas kepada penggunanya, memungkinkan para pengguna untuk memantau dan mengelola data lalu lintas dengan mudah tanpa terikat oleh perangkat yang mereka gunakan.

Berdasarkan hasil penelitian yang telah dilakukan, disarankan agar peluang pengembangan layanan *web service* dengan memanfaatkan kerangka kerja Laravel dan juga *micro framework* Lumen dipertimbangkan, mengingat perkembangan teknologi yang terus berlangsung. Dalam konteks penelitian ini, diharapkan bahwa layanan *web service* yang dikembangkan akan memberikan kontribusi dalam pengembangan berikutnya, khususnya ketika diintegrasikan dengan berbagai *platform* yang berbeda.

DAFTAR PUSTAKA

- [1] DPR, "UU No. 22 Tahun 2009 Tentang Lalu Lintas dan Angkutan Jalan," 2009. https://www.dpr.go.id/dokjdih/document/uu/UU_2009_22.pdf (diakses 9 Juni 2023).
- [2] Sulistyowati, A., dan Muazansyah, I. Optimalisasi Pengelolaan dan Pelayanan Transportasi Umum (Studi pada "Suroboyo Bus" di Surabaya). Iapa Proceedings Conference, 152-165, 2019.
- [3] Insan, I. H., Abid, A., Istianah, I., Lerick W, M., dan Kasturi S, R. Implementasi Prinsip Green Government pada Pemerintah Daerah Kota Bogor, 2023.
- [4] Azizah, A. N., Budiharjo, A., dan Maimunah, S. Kajian Manajemen Lalu Lintas di Kawasan Pasar Bogor. Vol. 23, No. 1, April 2022, Hal. 01~08, 2022.
- [5] Nugroho, A. B. Sistem Pemantau Pelanggaran Lalu Lintas Pada Traffic Light Menggunakan Sensor SRF04 dan Kamera Berbasis Mikrokontroler AtMega8535. Vol. 1, No. 1, Agustus 2019.
- [6] Baharuddin, H. Wakkang, dan Irianto, B. Implementasi *web service* dengan Metode REST API untuk Integrasi Data COVID-19 di Sulawesi Selatan. Jurnal Sintaks Logika, Vol. 2, No. 1, Januari – 2022.
- [7] Pangestu, H. A., Kurniadi, D., dan Septiana, Y. Aplikasi Pengelolaan Data Pegawai Berbasis REST API untuk Transfer Data Real time dengan *framework* Codeigniter. Vol. 19, No. 1, Hal 313-322, 2022.
- [8] Paramitha, I. A. K. P., Wiharta, D. M., dan Suyadnya, I. M. A. Perancangan dan Implementasi RESTful API pada Sistem Informasi Manajemen Dosen Universitas Udayana. Jurnal Spektrum, Vol. 9, No. 3, September 2022.
- [9] Iqbal, M., dan Nurwati. (2023). Penerapan Sistem Terintegrasi Menggunakan RESTful API pada Dealer Management System Panca Niaga Sei Piring. Vol. VI (1), Feb 2023.
- [10] Lesmana, R.F., dan Razaq, J.A. Sistem Penelitian dan Pengabdian Masyarakat dengan Integrasi Data Akademik menggunakan REST API. Jurnal Volume 6, Nomor 1, Januari 2023.
- [11] Darmawan, D., Kusumah, F.S.F., dan Al Ikhsan, S.H. *web service* untuk Transaksi Data pada Aplikasi Fasilitas Keuangan dengan Metode REST. Jurnal Volume 5 Nomor 2, September 2021.
- [12] Rezki, M., Ihsan, M. I. R., dan Nurdiani, S. Perancangan Sistem Informasi Pengolahan Data Transaksi pada Toko Kelontong Tebas Berbasis *web*, Maret 2020.
- [13] Karim, A. S., dan A'isy, L. R. Sistem Informasi Lalu Lintas di Kota Bandar Lampung Berbasis CCTV, Juni 2019.
- [14] Laravel. (n.d.). "Installation - Lumen - PHP *Micro-framework*." 2023 <https://lumen.laravel.com/docs> (Diakses pada 6 Juli 2023)
- [15] Masse, Mark. REST API Design Rulebook, 2011.
- [16] Oracle, "MySQL - *The world's most popular open source database*," <https://www.oracle.com/mysql/what-is-mysql/> (diakses 12 Juli 2023)
- [17] R. S. Pressman, *Software Engineering: A Practitioner's Approach*, 7 ed. New York: McGraw-Hill, 2010.