

Sistem Monitoring Indeks Kualitas Udara Berbasis Open Meteo Air Quality API

I Gede Surya Rahayuda^{a1}, Ni Putu Linda Santiari^{a2}

^aProgram Studi Informatika, Fakultas MIPA, Universitas Udayana
Jalan Kampus Bukit Jimbaran, Bali, Indonesia
¹igedesuryarahayuda@unud.ac.id (Corresponding author)

^bProgram Studi Sistem Informasi, Fakultas Informatika dan Komputer, ITB STIKOM Bali
Jalan Raya Puputan No. 86 Renon, Denpasar, Bali, Indonesia
²linda_santiari@stikom-bali.ac.id

Abstract

This research develops an air quality monitoring system utilizing the Open Meteo API and OpenStreetMap API, employing the Software Development Life Cycle (SDLC). Users input latitude and longitude to access the air quality index (AQI) at a specific location, with a default one-day hourly forecast. Configuration options include location and forecast duration. The system successfully determines AQI values based on user settings and employs color-coded visual representations following US Air Quality Index standards for easier comprehension of health risks. The SDLC ensures a streamlined development process. Additionally, the OpenStreetMap API aids in geographic coordinate-based location identification. Usability evaluation, conducted through Google Analytics and a Google Form questionnaire, indicates high user satisfaction, particularly regarding information clarity, content completeness, and overall experience. Future enhancements may target optimizing page loading speed and increasing user engagement. In summary, this research contributes to accessible and understandable air quality monitoring, benefiting users seeking reliable AQI data.

Keywords: AQI, API, Open Meteo, Vercel, GitHub, Serverless Architecture.

1. Pendahuluan

Dalam era modern yang dipenuhi inovasi teknologi, pemantauan kualitas udara menjadi semakin penting untuk menjaga kesehatan dan kesejahteraan masyarakat. Kualitas udara yang buruk dapat memiliki dampak serius terhadap kesehatan manusia, terutama bagi kelompok rentan seperti anak-anak, lansia, dan individu dengan masalah pernapasan. Oleh karena itu, pengembangan sistem monitoring kualitas udara menjadi suatu kebutuhan. Penelitian ini mengembangkan solusi yang inovatif dengan mengintegrasikan Open Meteo API dan OpenStreetMap API untuk memberikan informasi kualitas udara berbasis lokasi. Melalui penggunaan data latitude dan longitude, sistem ini memungkinkan pengguna untuk dengan mudah melihat indeks kualitas udara (AQI) pada suatu tempat tertentu [1] [2]. Keunggulan sistem ini tidak hanya terletak pada kemampuannya memberikan informasi secara akurat, tetapi juga pada fleksibilitasnya dalam mengizinkan pengguna untuk mengonfigurasi lokasi dan masa depan. Dengan demikian, pengguna dapat dengan lebih tepat merencanakan aktivitas mereka berdasarkan kualitas udara di suatu wilayah. Selain itu, sistem ini menggunakan arsitektur serverless yang memiliki keunggulan efisiensi biaya, scalability, dan reliability. Efisiensi biaya karena sistem hanya perlu membayar biaya untuk sumber daya yang digunakan. Scalability karena sistem dapat dengan mudah diskala untuk menangani jumlah data yang lebih besar atau beban kerja yang lebih berat. Reliability karena sistem memiliki tingkat keandalan yang tinggi. Terakhir, sistem ini dapat diintegrasikan dengan GitHub untuk memudahkan pengembangan dan pengelolaan sistem. Dimana beberapa kelebihan tersebut merupakan suatu pengembangan dari hasil penelitian sebelumnya. Penggunaan Software Development Life Cycle (SDLC) dalam pengembangan sistem ini memberikan dasar yang kokoh, memastikan bahwa proses pengembangan dilakukan dengan metode yang terstruktur dan terorganisir. Sistem ini juga mengadopsi standar United States Air Quality Index (AQI) untuk kategorisasi hasil pengukuran, memberikan informasi yang mudah dimengerti oleh pengguna

melalui representasi visual menggunakan warna. Dengan melibatkan OpenStreetMap API, sistem ini juga meningkatkan kemampuan identifikasi lokasi berdasarkan koordinat geografis. Inovasi ini memberikan dimensi tambahan pada pengalaman pengguna, menjadikan sistem ini tidak hanya sebagai alat pemantauan kualitas udara, tetapi juga sebagai panduan geografis yang komprehensif. Melalui pendekatan yang holistik, penelitian ini diharapkan dapat memberikan kontribusi positif dalam memahami dan mengatasi permasalahan kualitas udara di berbagai wilayah. Dengan menggunakan teknologi terkini, sistem monitoring ini tidak hanya menjadi alat penting bagi individu, tetapi juga untuk otoritas kesehatan dan lingkungan dalam pengambilan keputusan yang berbasis data [3] [4].

2. Tinjauan Pustaka

Berikut analisis review pada beberapa artikel terkait penelitian yang dilakukan. Artikel "**A Review on Air Quality Indexing System**" dalam Asian Journal of Atmospheric Environment oleh Kanchan, Amit Kumar Gorai, dan Pramila Goyal pada tahun 2015 membahas pentingnya Indeks Kualitas Udara (AQI) atau Air Pollution Index (API) dalam memberikan informasi tentang tingkat keparahan pencemaran udara kepada masyarakat. Meskipun terdapat berbagai metode yang dikembangkan oleh peneliti dan lembaga lingkungan hidup, kekurangan metodologi yang diterima secara universal terjadi. Makalah tersebut menyajikan tujuan utama AQI atau API, yaitu mengidentifikasi wilayah dengan kualitas udara buruk dan memberikan informasi kepada masyarakat tentang tingkat keparahan paparan. Indeks ini dapat dikategorikan sebagai polutan tunggal atau multipolutan, masing-masing dengan kekuatan dan kelemahan sendiri. Penelitian tersebut bertujuan memberikan tinjauan komprehensif terhadap indeks kualitas udara di seluruh dunia [5]. Sementara itu, makalah "**A Case Study of Web API Evolution**" yang dipresentasikan pada IEEE World Congress on Services 2015 oleh S.M. Sohan, Craig Anslow, dan Frank Maurer, membahas tantangan integrasi aplikasi menggunakan API web. Artikel tersebut menyoroti potensi perubahan dalam API web yang dapat mengganggu aplikasi yang bergantung, mencakup kurangnya dukungan untuk versi API lama, dokumentasi yang tidak memadai, dan komunikasi perubahan yang tidak efektif. Melalui studi kasus evolusi API web, penelitian ini memberikan rekomendasi bagi praktisi dan peneliti mengenai profil perubahan API, strategi versi, serta pendekatan dokumentasi dan komunikasi yang efektif [6]. Selanjutnya, "**Serverless Architecture - A Revolution in Cloud Computing**" oleh R. Arokia Paul Rajan pada Tenth International Conference on Advanced Computing (ICoAC) 2018 memberikan gambaran komprehensif tentang komputasi tanpa server. Artikel tersebut membahas prinsip dasar komputasi tanpa server, membandingkan platform utama seperti AWS Lambda, Google Cloud Functions, Microsoft Azure Functions, dan IBM Cloud Functions, serta mengeksplorasi kasus penggunaan dunia nyata. Artikel ini juga menyoroti tantangan teknis dan organisasi, memberikan wawasan tentang tren masa depan, dan mengakui potensi dampak komputasi tanpa server pada pengembangan perangkat lunak di berbagai industri. Artikel ini menyajikan konten yang terstruktur dengan baik dan bermanfaat bagi para peneliti, pengembang, dan organisasi yang tertarik pada komputasi tanpa server [7]. Secara keseluruhan, ketiga jurnal tersebut memberikan kontribusi yang signifikan dalam pengembangan teknologi atau metode Sistem Monitoring Indeks Kualitas Udara berbasis Open Meteo Air Quality API.

3. Metode Penelitian

Dalam pelaksanaannya, penelitian ini mengaplikasikan pendekatan Software Development Life Cycle (SDLC) sebagai kerangka kerja yang terstruktur untuk memastikan kelancaran pengembangan sistem. Proses SDLC mencakup serangkaian tahapan, mulai dari perencanaan yang komprehensif, analisis mendalam, desain yang terfokus, implementasi yang cermat, hingga pemeliharaan sistem yang berkelanjutan. Pendekatan ini memberikan fondasi kokoh untuk mengelola setiap langkah pengembangan dengan disiplin dan kerapian, sehingga diharapkan mampu mencapai tujuan penelitian dengan efektif dan efisien [8] [9].

3.1. Perencanaan

Dalam tahap perencanaan, penelitian ini mengidentifikasi kebutuhan sistem berbasis kualitas udara dan fungsionalitas pengguna dengan cermat. Langkah-langkah ini melibatkan penentuan sumber daya yang diperlukan, termasuk strategi penggunaan Open Meteo API dan OpenStreetMap API untuk memastikan akuisisi data yang tepat dan komprehensif. Proses ini diarahkan pada pemahaman yang mendalam terhadap spesifikasi sistem guna memenuhi kebutuhan pengguna serta memastikan integrasi yang efisien dengan sumber daya eksternal yang relevan.

3.2. Analisis

Dalam fase analisis, penelitian ini melakukan evaluasi mendalam terhadap kebutuhan pengguna dengan tujuan menentukan fitur utama sistem. Proses ini melibatkan analisis kebutuhan pengguna, khususnya dalam konteks konfigurasi lokasi dan masa depan. Selain itu, tahap ini mencakup penyusunan spesifikasi sistem yang jelas dan merinci serta penentuan data yang dibutuhkan dari API eksternal, seperti Open Meteo dan OpenStreetMap. Hasil dari analisis ini akan menjadi landasan untuk perancangan sistem yang responsif dan sesuai dengan ekspektasi pengguna.

3.3. Desain

Dalam proses desain, penelitian ini mengadopsi pendekatan berbasis pseudocode untuk merinci struktur alur kerja sistem dengan lebih terperinci. Pseudocode digunakan sebagai representasi abstrak dari langkah-langkah dan logika program, memberikan deskripsi secara lebih rinci tentang implementasi setiap tahap, dari penerimaan input hingga penentuan kategori Air Quality Index (AQI). Pseudocode memberikan kejelasan dalam merancang logika sistem, memastikan bahwa setiap langkah diimplementasikan dengan akurat dan efisien sesuai dengan analisis kebutuhan pengguna. Dengan demikian, desain pseudocode ini menjadi dasar untuk implementasi yang berhasil dan sesuai dengan spesifikasi yang telah ditetapkan.

3.4. Implementasi

Dalam tahap implementasi, penelitian ini memanfaatkan HTML, CSS, dan JavaScript untuk mentransformasikan desain menjadi kode sumber yang fungsional. Penerapan ini juga melibatkan penggunaan Bootstrap untuk meningkatkan responsivitas dan estetika tampilan sistem. Selanjutnya, untuk mempublikasikan sistem secara online, dilakukan hosting web menggunakan platform Vercel yang terintegrasi dengan GitHub. Pendekatan ini tidak hanya memastikan ketersediaan sistem secara daring tetapi juga memudahkan pengelolaan kode sumber dan pembaruan melalui repositori GitHub. Dengan implementasi ini, diharapkan sistem dapat diakses dengan mudah dan memberikan pengalaman pengguna yang optimal.

3.5. Pengujian

Dalam fase pengujian, penelitian ini menerapkan pengujian black box, yang berfokus pada fungsionalitas sistem tanpa memperhatikan detail implementasi internal. Pengujian ini bertujuan untuk memverifikasi kemampuan sistem dalam memberikan hasil yang akurat sesuai dengan konfigurasi pengguna. Dengan memanfaatkan metode ini, pengujian dapat dilakukan dengan pendekatan yang lebih holistik, menguji berbagai kasus pengguna tanpa bergantung pada struktur internal sistem. Hasil dari pengujian black box ini menjadi indikator kehandalan dan kesesuaian sistem dalam menghadapi situasi nyata, sehingga memastikan kualitas dan performa yang diinginkan.

3.6. Pemeliharaan

Dalam tahap pemeliharaan (2.7), penelitian ini berfokus pada tindakan menjaga dan memperbarui sistem agar tetap relevan dan berkinerja optimal. Hal ini melibatkan pemantauan dan adaptasi terhadap perubahan pada API eksternal, seperti Open Meteo dan OpenStreetMap, untuk memastikan ketersediaan data yang akurat. Selain itu, respons terhadap umpan balik pengguna menjadi bagian integral dari pemeliharaan ini, dengan tujuan untuk meningkatkan pengalaman pengguna dan mengidentifikasi serta memperbaiki bug yang mungkin terjadi. Dengan pendekatan ini, sistem dapat terus berkembang sesuai dengan kebutuhan pengguna dan lingkungan eksternalnya.

4. Hasil dan Diskusi

Pengembangan sistem ini menggunakan pendekatan Software Development Life Cycle (SDLC) sebagai metodologi inti untuk memastikan bahwa seluruh proses pengembangan berjalan sesuai dengan tahapan yang terstruktur dan terorganisir. Dengan menerapkan SDLC, setiap fase dalam siklus pengembangan, mulai dari perencanaan, analisis, desain, implementasi, hingga pengujian dan pemeliharaan, dikelola dengan cermat untuk memastikan keberhasilan dan kehandalan sistem.

4.1. Perencanaan

Dalam tahap perencanaan penelitian ini menghasilkan identifikasi kebutuhan sistem yang didasarkan pada analisis kebutuhan pengguna, dengan fokus utama pada informasi kualitas udara dan fungsionalitas pengguna. Kebutuhan sistem mencakup kemampuan untuk melihat indeks kualitas udara (AQI) berdasarkan lokasi geografis, memungkinkan konfigurasi latitude dan longitude, serta

memberikan perkiraan cuaca dalam rentang waktu tertentu. Selanjutnya, hasil penentuan sumber daya menetapkan penggunaan Open Meteo API dan OpenStreetMap API sebagai sumber daya utama. Open Meteo API digunakan untuk mengakses data kualitas udara dan perkiraan cuaca, sedangkan OpenStreetMap API digunakan untuk mengidentifikasi nama tempat berdasarkan koordinat geografis. Integrasi kedua API ini menjadi fondasi sistem monitoring indeks kualitas udara untuk menyediakan informasi yang akurat dan relevan kepada pengguna.

4.2. Analisis

Berdasarkan perencanaan yang telah disusun, diperoleh suatu analisis yang menunjukkan bahwa sistem ini membutuhkan data array dari API Air Quality Open Meteo yang dapat diakses melalui penggunaan JavaScript. Data nama lokasi kemudian diperoleh berdasarkan data array [Open Meteo](#), yang selanjutnya diolah menggunakan [Open Street Map](#) untuk memperoleh informasi lokasi yang lebih lengkap. Untuk mendukung fungsionalitas sistem, beberapa halaman yang esensial diperlukan, antara lain: `index.html` sebagai halaman utama, `script.js` untuk mengelola logika dan interaksi dengan API, serta `style.css` sebagai file stylesheet untuk penataan tampilan dan gaya halaman web. Dengan pengaturan ini, sistem diharapkan dapat beroperasi dengan baik, menyediakan informasi kualitas udara yang relevan, dan menampilkan hasilnya dengan tata letak yang bersih dan sesuai dengan desain yang diinginkan.

Klasifikasi tingkat AQI menggunakan United States Air Quality Index (AQI) memberikan informasi tentang kualitas udara di suatu daerah berdasarkan rentang nilai yang diperoleh. Rentang nilai ini dapat memberikan gambaran sejauh mana kualitas udara di suatu lokasi, dan masing-masing rentang tersebut memiliki kategori kesehatan yang berbeda. Berikut adalah penjelasan lebih lanjut tentang setiap kategori:

- **Good (0-50):** Rentang ini menunjukkan bahwa kualitas udara dianggap baik, dengan risiko polusi udara yang rendah atau hampir tidak ada. Udara dalam kategori ini dianggap aman untuk pernapasan oleh semua kelompok. Warna umumnya adalah hijau, mencerminkan kondisi yang aman dan baik untuk pernapasan.
- **Moderate (51-100):** Kategori ini menandakan bahwa kualitas udara adalah sedang, dan meskipun secara umum masih aman, beberapa polutan udara mungkin menyebabkan risiko kecil bagi individu yang sangat peka terhadap polusi udara. Warna umumnya adalah kuning atau kuning muda, menandakan tingkat peringatan yang lebih tinggi, meskipun masih dalam kisaran yang dapat diterima.
- **Unhealthy for Sensitive Groups (101-150):** Rentang ini mengindikasikan bahwa individu yang termasuk dalam kelompok sensitif, seperti mereka yang memiliki kondisi pernapasan atau jantung, anak-anak, dan orang dewasa yang lebih tua, mungkin mengalami efek kesehatan jika terpapar dalam waktu yang lama. Warna umumnya adalah oranye, memberikan peringatan terhadap risiko kesehatan bagi kelompok rentan.
- **Unhealthy (151-200):** Kategori ini menunjukkan bahwa kualitas udara menjadi tidak sehat, dan semua orang dapat mulai mengalami efek kesehatan. Individu yang termasuk dalam kelompok sensitif mungkin mengalami efek yang lebih serius. Warna umumnya adalah merah. Ini menunjukkan bahwa semua orang, terlepas dari kondisi kesehatan, mungkin mengalami efek yang lebih serius.
- **Very Unhealthy (201-300):** Rentang ini menggambarkan bahwa kualitas udara sangat tidak sehat, dan populasi umum dapat mulai mengalami efek kesehatan yang serius. Semua orang, terlepas dari kondisi kesehatan, dapat terpengaruh. Warna umumnya adalah ungu. Ini mencerminkan tingkat peringatan yang tinggi dan memerlukan tindakan pencegahan serius untuk melindungi kesehatan masyarakat.
- **Hazardous (301-500):** Kategori ini merupakan tingkatan paling tinggi dalam skala AQI dan menunjukkan kondisi kesehatan yang darurat. Kualitas udara dalam rentang ini dapat menyebabkan efek kesehatan yang serius pada semua orang. Tindakan darurat mungkin diperlukan. Warna umumnya adalah merah tua atau coklat, memberikan peringatan yang sangat serius dan menunjukkan adanya bahaya kesehatan yang mendesak. Tindakan darurat mungkin diperlukan [4].

4.3. Desain

Dalam fase desain sistem, pendekatan pseudocode berhasil diterapkan untuk merinci struktur dan logika alur kerja sistem. Pseudocode memberikan pandangan yang lebih terperinci terkait implementasi

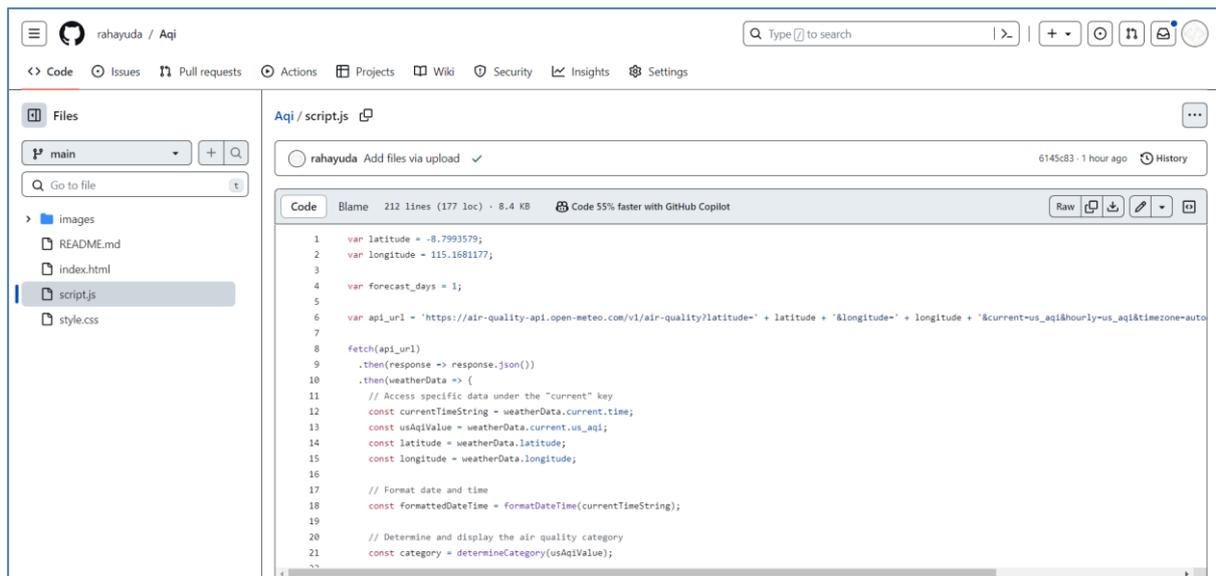
setiap langkah, mulai dari penerimaan input hingga penentuan kategori Air Quality Index (AQI). Hasil desain ini menciptakan landasan yang jelas dan terstruktur untuk implementasi selanjutnya, memastikan bahwa setiap aspek dari sistem diwujudkan dengan akurat dan efisien. Arsitektur serverless, sebagai pendekatan infrastruktur yang dapat diintegrasikan, memberikan landasan yang ideal untuk menerapkan pseudocode tersebut. Dalam arsitektur serverless, sistem dibangun tanpa kebutuhan untuk mengelola infrastruktur server secara langsung. Sebaliknya, fungsi-fungsi kecil yang menyusun alur kerja diimplementasikan sebagai layanan terpisah. Setiap fungsi serverless bertanggung jawab untuk tugas tertentu, seperti penerimaan input, pemrosesan data, dan penentuan kategori AQI. Kelebihan arsitektur serverless terletak pada fleksibilitas dan skalabilitasnya. Fungsi-fungsi serverless dapat diaktifkan secara otomatis ketika diperlukan dan dinonaktifkan saat tidak digunakan, mengoptimalkan penggunaan sumber daya dan mengurangi biaya infrastruktur. Integrasi arsitektur serverless dengan pendekatan pseudocode membantu mencapai konsistensi, keberlanjutan, dan efisiensi dalam pengembangan sistem Monitoring Indeks Kualitas Udara [10].

1. Inisialisasi variabel: latitude, longitude, forecast_days, dan api_url.
2. Ambil data cuaca menggunakan api_url dan proses data:
 - a. Ekstrak informasi yang relevan.
 - b. Tampilkan kategori kualitas udara saat ini, nilai, koordinat, dan warna latar belakang.
 - c. Isi tabel ramalan.
 - d. Tangani kesalahan jika ada.
3. Implementasikan fungsi updateData:
 - a. Dapatkan dan validasi nilai input baru.
 - b. Perbarui variabel dan api_url.
 - c. Ulangi langkah 2a hingga 2d.
 - d. Tutup 'inputModal'.
 - e. Ambil dan tampilkan data lokasi.
4. Implementasikan fungsi fetchLocationData:
 - a. Ambil data lokasi.
 - b. Tampilkan nama lokasi.
5. Implementasikan berbagai fungsi bantu:
 - a. formatDate untuk format tanggal dan waktu.
 - b. padZero untuk menambahkan nol di depan.
 - c. determineCategory untuk menentukan kategori AQI.
 - d. determineCaution untuk mendapatkan informasi peringatan.
 - e. setBackgroundColor untuk mengatur warna latar belakang.
 - f. populateForecastTable untuk mengisi tabel ramalan.
6. Tampilkan data pada halaman web.
7. Implementasikan desain responsif menggunakan Bootstrap.
8. Hosting halaman web di Vercel untuk akses publik:
<https://aqipollution.vercel.app>.
9. Lakukan pengujian black box untuk memastikan hasil yang akurat.
10. Jaga dan perbarui sistem untuk menyesuaikan perubahan API eksternal, tanggap umpan balik pengguna, dan perbaiki bug yang teridentifikasi.

4.4. Implementasi

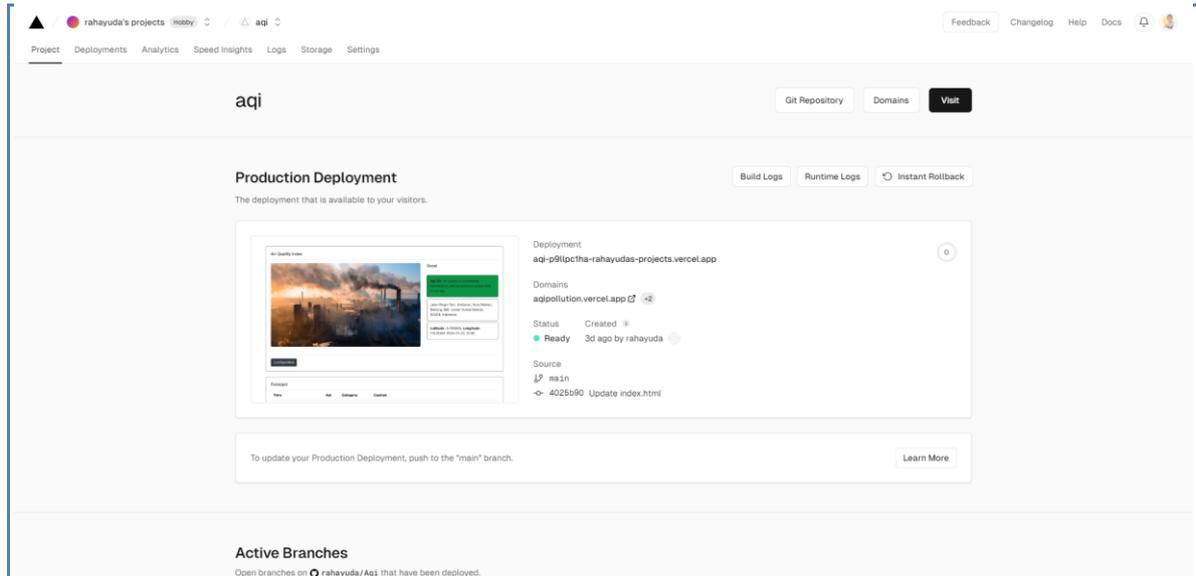
Sebagai bagian krusial dari siklus pengembangan perangkat lunak, tahap implementasi melibatkan transformasi desain konseptual menjadi suatu produk yang dapat berjalan. Dalam konteks Sistem Monitoring Indeks Kualitas Udara, implementasi dilakukan dengan menggunakan sejumlah teknologi kunci. HTML, CSS, dan JavaScript digunakan untuk menerjemahkan desain sistem ke dalam kode sumber yang dapat dijalankan secara efektif. Web hosting diberdayakan melalui Vercel, platform yang terintegrasi dengan GitHub, memungkinkan penerapan praktis dan manajemen mudah dari kode sumber. Bootstrap, kerangka kerja front-end yang terkenal, dimanfaatkan untuk meningkatkan responsivitas dan estetika antarmuka pengguna, memastikan pengalaman pengguna yang menyenangkan dan mudah diakses. Dengan kombinasi teknologi ini, tahap implementasi menjadi fondasi utama bagi fungsionalitas dan tampilan Sistem Monitoring Indeks Kualitas Udara.

Menerapkan desain ke dalam kode sumber Sistem Monitoring Indeks Kualitas Udara melibatkan integrasi HTML, CSS, dan JavaScript sebagai komponen utama. HTML digunakan untuk menentukan struktur dan elemen-elemen pada halaman web, CSS bertanggung jawab atas penataan dan tampilan estetika, sementara JavaScript memberikan fungsionalitas dinamis dan interaktivitas pada halaman. Dalam repository GitHub, terdapat tangkapan layar (screenshot) yang mencerminkan struktur folder dan berkas serta beberapa cuplikan kode krusial terlihat pada **gambar 1**. **index.html** merupakan berkas utama yang menentukan struktur dan konten dari halaman web. Di dalamnya, elemen-elemen HTML ditempatkan untuk membangun antarmuka pengguna. Struktur yang terorganisir memudahkan pemahaman dan pemeliharaan kode. **script.js** adalah berkas JavaScript yang memberikan logika dan interaktivitas pada halaman. Dengan menggunakan JavaScript, Sistem Monitoring Indeks Kualitas Udara dapat mengambil data dari API, memproses informasi, dan mengupdate tampilan secara dinamis tanpa perlu me-refresh halaman. **style.css** berperan dalam mendefinisikan tata letak dan tampilan estetika elemen-elemen HTML. Melalui CSS, warna, ukuran, dan posisi elemen dapat diatur untuk menciptakan antarmuka pengguna yang menarik dan mudah dibaca.



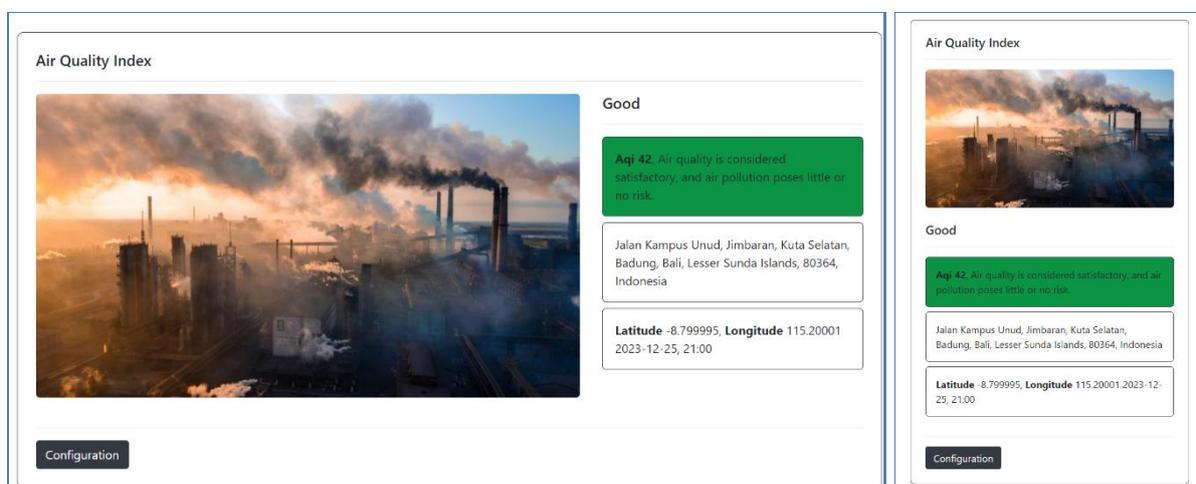
Gambar 1. Repositori Kode pada Github

Proses hosting web merupakan langkah krusial dalam menghadirkan sebuah proyek secara daring. Dalam konteks Sistem Monitoring Indeks Kualitas Udara, proyek ini dihosting menggunakan platform Vercel yang terintegrasi secara langsung dengan GitHub. Integrasi ini mempermudah pengelolaan proyek dan penerapan perubahan, membuatnya menjadi pilihan yang efisien bagi pengembang. Vercel adalah platform cloud yang memungkinkan pengguna untuk menyimpan dan mendeploy proyek web secara cepat dan mudah. Dengan integrasi GitHub, setiap kali terjadi perubahan pada repository GitHub, Vercel secara otomatis memicu proses pembangunan (build) dan merilis (deploy) versi terbaru dari proyek. Hal ini memastikan bahwa proyek selalu tersedia dalam kondisi terkini. Keuntungan utama dari integrasi Vercel dan GitHub adalah efisiensi pengelolaan kode sumber dan implementasi perubahan. Proses deployment yang otomatis menghilangkan kebutuhan untuk melakukan langkah-langkah manual, sehingga pengembang dapat lebih fokus pada pengembangan fitur dan fungsionalitas proyek seperti terlihat pada **gambar 2**. Selain itu, Vercel menyediakan URL unik yang mudah diingat (<https://aqipollution.vercel.app>), memudahkan pengguna untuk mengakses Sistem Monitoring Indeks Kualitas Udara secara daring dengan cepat.

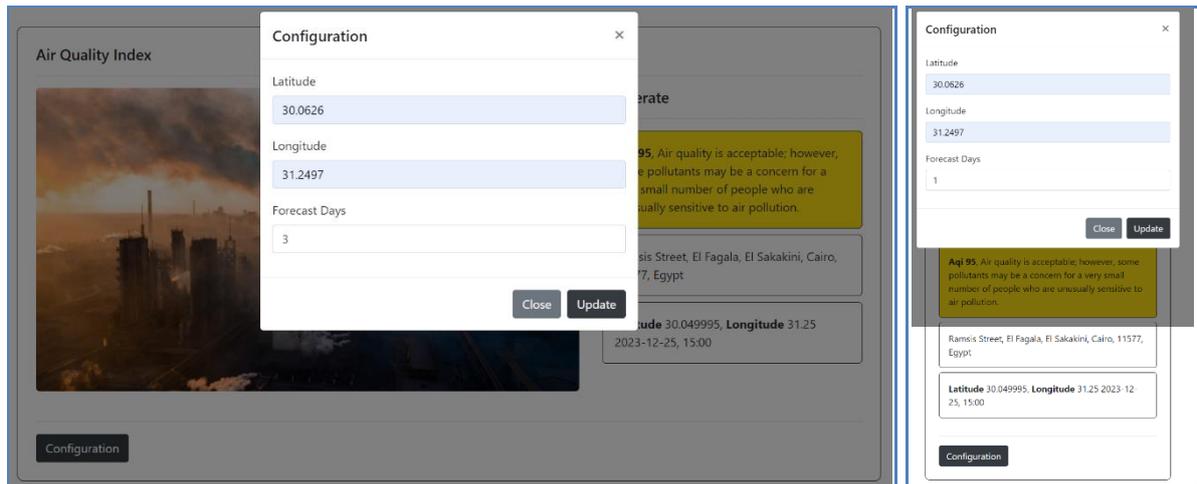


Gambar 2. Vercel Deployment

Dalam pengembangan Sistem Monitoring Indeks Kualitas Udara, responsivitas dan tampilan yang estetik menjadi fokus penting. Untuk mencapai hal ini, proyek ini memanfaatkan Bootstrap, sebuah kerangka kerja CSS yang terkenal, untuk merancang antarmuka yang responsif dan menarik. Bootstrap menyediakan berbagai komponen dan utilitas yang mempermudah pengembangan tata letak yang responsif tanpa harus menulis banyak kode CSS kustom. Tampilan web versi desktop dari Sistem Monitoring Indeks Kualitas Udara mencerminkan desain yang bersih dan terorganisir. Pengguna dapat dengan mudah membaca informasi mengenai kategori indeks kualitas udara, nilai AQI, dan detail lokasi. Tata letak ini memberikan pengalaman pengguna yang intuitif dan informatif. Selain itu, responsivitas proyek ini dijamin melalui penggunaan Bootstrap, yang mengoptimalkan tampilan web saat diakses melalui perangkat mobile. Dengan demikian, pengguna dapat mengakses Sistem Monitoring Indeks Kualitas Udara dengan lancar tanpa mengalami kesulitan tata letak atau keterbacaan, seperti terlihat pada **gambar 3**. Sistem konfigurasi juga dihadirkan melalui modal yang dapat diakses dengan mudah. Modal ini memungkinkan pengguna untuk memasukkan data lokasi baru dan jumlah hari perkiraan cuaca dengan nyaman, terlihat pada **gambar 4**. Tampilan yang bersih dan sederhana memberikan pengalaman pengguna yang lebih baik dalam mengonfigurasi Sistem Monitoring Indeks Kualitas Udara



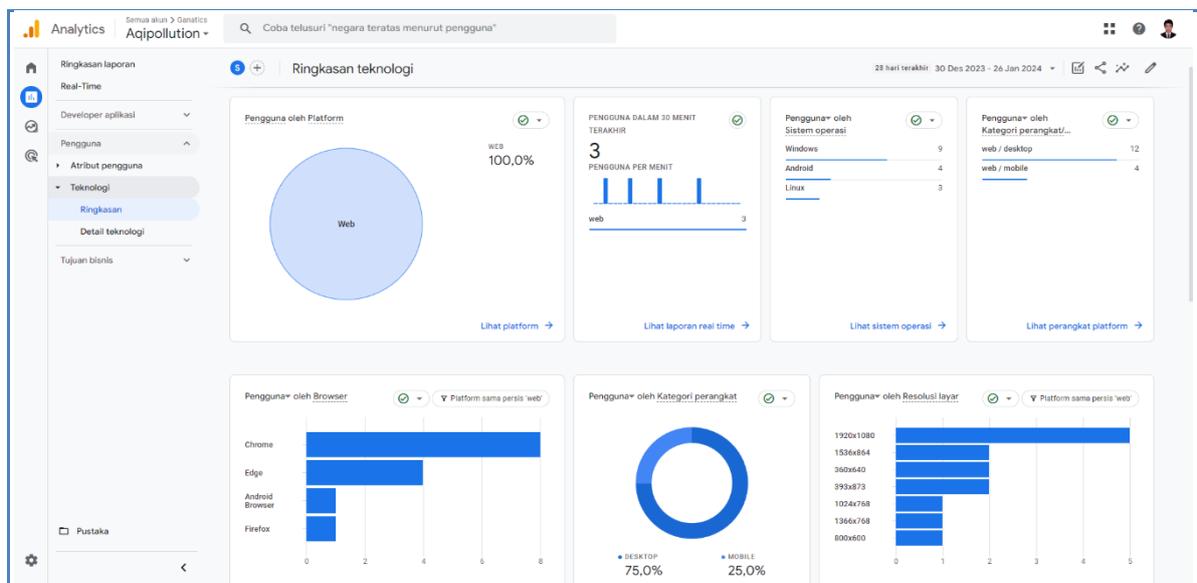
Gambar 3. Tampilan Desktop dan Mobile



Gambar 4. Tampilan Konfigurasi

4.5. Pengujian

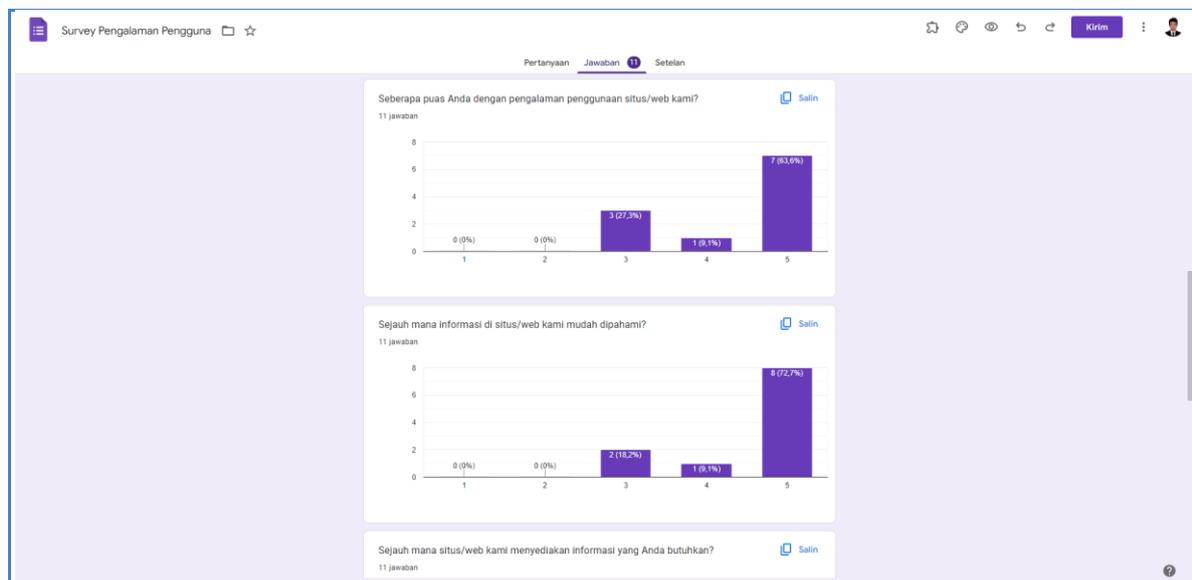
Selama periode evaluasi dari tanggal 23 Januari hingga 27 Januari 2024, statistik penggunaan yang tercatat dalam Google Analytics menunjukkan pola yang menarik seperti terlihat pada gambar 5. Mayoritas pengguna menggunakan sistem operasi Windows dengan persentase 39.13%, diikuti oleh pengguna Android sebesar 17.39%, dan Linux sebesar 13.04%. Dari segi perangkat, sebagian besar akses dilakukan melalui Web/Desktop dengan persentase 52.17%, sementara penggunaan Web/Mobile mencapai 17.39%. Chrome adalah browser yang paling banyak digunakan dengan persentase 34.78%, diikuti oleh Edge dengan 17.39%, dan Android Browser hanya mencapai 4.35%. Resolusi layar 1920x1080 mendominasi dengan 21.74%, sementara 1536x864 dan 360x640 masing-masing mencapai 8.70%. Data penggunaan juga mencatat Sesi Engagement mencapai 100%, menunjukkan tingkat keterlibatan yang tinggi dari pengguna, dengan Rasio Engagement sebesar 61.76%, menunjukkan sebagian besar pengguna terlibat dalam interaksi. Waktu Engagement Rata-Rata adalah 46 detik, menandakan sebagian besar pengguna menghabiskan waktu singkat di situs, sementara Jumlah Peristiwa mencapai 236, menunjukkan aktivitas yang cukup signifikan dalam interaksi pengguna dengan situs.



Gambar 5. Google Analytics

Pada tahap pengujian sistem menggunakan metode Black Box, fokus utamanya adalah pada pengujian fungsionalitas tanpa memperhatikan detail implementasi internal. Pengujian ini bertujuan untuk memverifikasi kemampuan Sistem Monitoring Indeks Kualitas Udara dalam memberikan hasil yang

akurat sesuai dengan konfigurasi pengguna yang diberikan. Hasil pengujian Black Box menunjukkan bahwa semua fitur pada sistem telah berjalan dengan baik dan sesuai dengan harapan. Sistem berhasil memberikan respons yang tepat dan akurat terhadap konfigurasi yang diberikan, termasuk pembaruan lokasi, penggunaan Open Meteo API, dan integrasi OpenStreetMap API. Pengujian ini mencakup evaluasi terhadap kemampuan sistem dalam menampilkan indeks kualitas udara (AQI) berdasarkan lokasi geografis, konfigurasi latitude dan longitude, serta perkiraan cuaca dalam rentang waktu tertentu. Semua aspek ini telah diuji secara menyeluruh, dan hasilnya memastikan bahwa Sistem Monitoring Indeks Kualitas Udara berfungsi dengan baik sesuai dengan kebutuhan pengguna. Dengan demikian, hasil pengujian Black Box memberikan keyakinan bahwa Sistem Monitoring Indeks Kualitas Udara siap untuk digunakan secara luas, memberikan informasi yang akurat dan bermanfaat terkait kualitas udara sesuai dengan lokasi dan preferensi pengguna.



Gambar 6. Kuisisioner

Berdasarkan hasil evaluasi menggunakan kuisisioner Google Form dengan skala Likert dari 1 hingga 5 seperti terlihat pada **gambar 6**, yang menilai lima aspek berbeda dari pengalaman penggunaan situs/web, diperoleh persentase tingkat kepuasan pengguna sebagai berikut: untuk Kepuasan Umum (P1), persentasenya sekitar 88.89%; untuk Kejelasan Informasi (P2), sekitar 93.33%; untuk Kelengkapan Konten (P3), sekitar 91.11%; untuk Kecepatan Memuat Halaman (P4), sekitar 88.89%; dan untuk Loyalitas (P5), sekitar 86.67%. Hasil ini menunjukkan adanya tingkat kepuasan yang tinggi dari pengguna terhadap berbagai aspek penggunaan situs/web, dengan skor tertinggi terdapat pada Kejelasan Informasi. Hal ini memberikan gambaran yang positif tentang pengalaman pengguna dan memberikan arahan untuk potensi perbaikan di masa depan.

Dari hasil evaluasi tersebut, dapat disimpulkan bahwa Sistem Monitoring Indeks Kualitas Udara telah berhasil dalam memberikan respons yang akurat dan memuaskan kepada pengguna. Namun, perlu dipertimbangkan untuk lebih memperhatikan tingkat kepuasan pada fitur tertentu yang mungkin mendapat penilaian lebih rendah, seperti kecepatan memuat halaman. Langkah-langkah untuk meningkatkan kepuasan pengguna pada area-area tertentu dapat dilakukan berdasarkan temuan ini.

4.6. Pemeliharaan

Pada tahap pemeliharaan, Sistem Monitoring Indeks Kualitas Udara akan terus dikembangkan dan diperbarui guna mengakomodasi perubahan pada API eksternal yang digunakan. Proses ini melibatkan respons terhadap umpan balik dari pengguna, dengan memperbaiki bug yang teridentifikasi dan meningkatkan fungsionalitas sistem sesuai kebutuhan. Selain itu, pemeliharaan sistem juga mencakup pemanfaatan layanan web hosting Vercel. Penggunaan Vercel sebagai platform hosting memastikan akses publik yang stabil dan responsif terhadap pengguna. Dengan demikian, sistem dapat diakses dengan baik oleh pengguna dari berbagai lokasi dan perangkat, menjaga ketersediaan informasi mengenai kualitas udara yang konsisten. Pemeliharaan ini merupakan langkah penting untuk menjaga performa optimal sistem seiring berjalannya waktu. Dengan terus melakukan pembaruan dan

perbaikan, Sistem Monitoring Indeks Kualitas Udara dapat terus memberikan layanan yang handal dan relevan bagi pengguna, serta tetap responsif terhadap perubahan lingkungan atau kebutuhan pengguna yang mungkin muncul.

5. Kesimpulan

Dalam penelitian ini, sebuah Sistem Monitoring Indeks Kualitas Udara berhasil dikembangkan dengan memanfaatkan pendekatan Software Development Life Cycle (SDLC) dan integrasi Open Meteo API serta OpenStreetMap API. Sistem ini memungkinkan pengguna untuk dengan mudah melihat Indeks Kualitas Udara (AQI) berdasarkan lokasi geografis, memberikan perkiraan cuaca dalam rentang waktu tertentu, dan konfigurasi lokasi serta masa depan. Melalui penerapan standar United States Air Quality Index (AQI) dalam representasi visual menggunakan warna, pengguna dapat dengan cepat memahami tingkat risiko kesehatan. Penggunaan SDLC memastikan proses pengembangan yang terstruktur, sementara integrasi OpenStreetMap API meningkatkan kemampuan identifikasi lokasi berdasarkan koordinat geografis. Responsif dan estetis, Sistem ini dihosting secara online menggunakan Vercel dan GitHub, memastikan ketersediaan dan manajemen kode yang efisien. Pengujian Black Box sukses menunjukkan kehandalan sistem, dan melalui pemeliharaan yang berkelanjutan, diharapkan sistem ini dapat terus memberikan informasi kualitas udara yang akurat dan dapat diandalkan kepada pengguna. Hasil evaluasi usability yang positif, seperti yang tergambar dari tingkat kepuasan yang tinggi dalam penggunaan situs/web, menegaskan kelayakan dan efektivitas sistem ini. Dengan pendekatan holistik, penelitian ini memberikan kontribusi positif dalam memahami dan mengatasi permasalahan kualitas udara, dengan memadukan teknologi terkini menjadi alat pemantauan yang accessible dan komprehensif.

Daftar Pustaka

- [1] World Health Organization, *Ambient Air Pollution: A Global Assessment of Exposure and Burden of Disease*. Geneva: WHO Press, 2018.
- [2] Y. Zhang, *Advances in Air Pollution Monitoring and Exposure Assessment*. FL: CRC Press, 2019.
- [3] U.S. Environmental Protection Agency (EPA), *Air Quality Index: A Guide to Air Quality and Your Health*. Washington, DC: EPA Publications, 2023.
- [4] Meteo France, *CAMS European air quality forecasts, ENSEMBLE data*. opernicus Atmosphere Monitoring Service (CAMS) Atmosphere Data Store (ADS), 2022.
- [5] Kanchan, A. K. Gorai, and P. Goyal, "A Review on Air Quality Indexing System," *Asian J. Atmos. Environ.*, 2015.
- [6] S. M. Sohan, C. Anslow, and F. Maurer, "A Case Study of Web API Evolution," *IEEE World Congr. Serv.*, 2015.
- [7] R. A. P. Rajan, "Serverless Architecture - A Revolution in Cloud Computing," *Tenth Int. Conf. Adv. Comput.*, 2018.
- [8] I. Sommerville, *Software Engineering*. Addison-Wesley, 2023.
- [9] Karl Wieggers, *Software Requirements*. Microsoft Press, 2023.
- [10] Syamsiah, "Perancangan Flowchart dan Pseudocode Pembelajaran Mengenal Angka dengan Animasi Untuk Anak PAUD Rambutan," *Satuan Tulisan Ris. dan Inov. Teknol.*, 2019.