



## Rancang Bangun Sistem Pemantauan Stasiun Cuaca Otomatis Berbasis *Augmented Reality* dengan Integrasi Web API dan Basis Data *Real-time*

Andi Tanri Seno Widyawati<sup>1</sup>, M. Zaky Zaim Muhtadi<sup>2\*</sup>

<sup>1-2</sup>Prodi Teknik Instrumentasi Kilang, Politeknik Energi dan Mineral Akamigas, Indonesia

\*Penulis Korespondensi: [zmuhtadi@esdm.go.id](mailto:zmuhtadi@esdm.go.id)

**Abstract.** *This study aims to design and develop an Augmented Reality (AR)-based application for real-time data monitoring of an automatic weather station at PT. XYZ. The system was developed using a prototyping method, involving five key stages: sensor data acquisition via CBOX devices and the Modbus RTU protocol, data transmission through a Flask-based Web API, storage in a MySQL database, QR code-based target image creation using the Vuforia Engine, and a 3D visualization interface developed with Unity integrated with the Vuforia SDK on Android devices. The system successfully received and stored weather data—such as temperature, humidity, and wind speed—into the MySQL database. The AR application also displayed a stable 3D interface panel over QR code markers, providing real-time data updates through an HTTP-based mechanism. This research demonstrates the successful integration of SCADA, Flask, MySQL, and Unity, enhancing both the functionality and commercial appeal of PT. XYZ's weather station products. Future research should focus on field testing, cloud network integration, and device compatibility.*

**Keywords:** *Augmented Reality; Automatic Weather Station ; Flask; Modbus RTU; Unity*

**Abstrak.** Penelitian ini bertujuan untuk merancang dan mengembangkan aplikasi berbasis Augmented Reality (AR) untuk pemantauan data secara real-time pada stasiun cuaca otomatis di PT. XYZ. Sistem ini dikembangkan menggunakan metode prototyping yang terdiri dari lima tahap utama: akuisisi data sensor melalui perangkat CBOX dan protokol Modbus RTU, transmisi data melalui Web API berbasis Flask, penyimpanan di database MySQL, pembuatan gambar target berbasis kode QR menggunakan Vuforia Engine, dan antarmuka visualisasi 3D yang dikembangkan dengan Unity yang terintegrasi dengan SDK Vuforia pada perangkat Android. Sistem berhasil menerima dan menyimpan data cuaca seperti suhu, kelembaban, dan kecepatan angin ke dalam database MySQL. Aplikasi AR juga berhasil menampilkan panel antarmuka 3D yang stabil di atas penanda kode QR dan menampilkan data secara real-time melalui mekanisme pengambilan data berbasis HTTP. Penelitian ini menunjukkan integrasi yang sukses antara SCADA, Flask, MySQL, dan Unity, yang meningkatkan fungsionalitas serta daya saing komersial produk stasiun cuaca otomatis PT. XYZ. Penelitian lanjutan disarankan untuk fokus pada uji lapangan, integrasi jaringan cloud, dan pengujian kompatibilitas perangkat.

**Kata kunci:** Kerangka Kerja Web; Kesatuan; Modbus RTU; Realitas Tertambah; Stasiun Cuaca Otomatis.

### 1. LATAR BELAKANG

PT. XYZ adalah perusahaan yang bergerak dibidang pengembangan sistem aplikasi otomasi dan digitalisasi. Salah satu produk yang dihasilkan adalah alat stasiun cuaca otomatis. Stasiun cuaca otomatis adalah alat yang mengukur parameter cuaca dengan sensor elektrik. Parameter cuaca tersebut terdiri dari radiasi matahari, suhu udara, tekanan udara, arah dan kecepatan angin, kelembapan udara, dan curah hujan. Pemantauan parameter-parameter cuaca diperlukan untuk mendapatkan informasi kondisi Wicaksana et al., (2021), yang saat ini dapat dioptimalkan melalui integrasi infrastruktur *internet of things* pada pengukurannya Widayanti & Fajarrohman, (a2024), sebagaimana halnya sistem pemantauan otomatis berbasis sensor

yang telah terbukti efektif untuk mengukur parameter lingkungan secara *real-time* (Dany Sucipto et al., 2026).

Sebagai perusahaan yang berfokus pada digitalisasi, PT. XYZ membutuhkan sebuah inovasi visualisasi untuk menciptakan sistem pemantauan yang tidak hanya fungsional, tetapi juga memiliki daya tarik visual yang tinggi. Visualisasi data yang menarik dan modern memiliki peranan strategis bukan sekadar karena urgensi teknis, melainkan sebagai nilai tambah (*added value*) yang membedakan produk perangkat stasiun cuaca otomatis PT. XYZ dari kompetitor di pasaran. Hal ini sejalan dengan temuan bahwa teknologi AR mampu meningkatkan nilai produk secara komersial melalui pengalaman pengguna yang lebih imersif dan interaktif (Venkat Ghodke et al., 2024).

Untuk merealisasikan daya tarik tersebut, teknologi *Augmented Reality* (AR) hadir sebagai solusi inovatif yang mampu mentransformasi data teknis menjadi visual 3D yang menarik dan intuitif. *Augmented Reality* adalah suatu konsep perpaduan antara *virtual reality* dengan *word reality*, sehingga obyek-obyek virtual 3 dimensi terlihat nyata dan menyatu dengan dunia nyata (Subari, 2014; Wicaksana et al., 2021). Kemampuan AR untuk menyajikan data sensor secara langsung di atas objek fisik menjadikannya antarmuka yang komprehensif untuk sistem instrumentasi industri Venkat Ghodke et al., (2024), termasuk dalam hal konsistensi kedalaman visualisasi objek virtual yang diproyeksikan ke lingkungan nyata (Seger et al., 2025).

Dengan memanfaatkan platform Unity, data parameter cuaca dari server dapat divisualisasikan secara *real-time* ke dalam antarmuka berbasis AR. Pendekatan ini memudahkan pengguna untuk memantau data kelembapan, suhu, hingga kecepatan angin langsung melalui perangkat *smartphone* yang diarahkan ke sebuah *marker*. (Nugroho & Pramono, (2017) dan Simon, (2023) telah membuktikan bahwa kombinasi Unity dan Vuforia merupakan platform yang handal untuk pengembangan aplikasi AR berbasis *marker* pada perangkat *mobile* Android.

Secara teknis, pengembangan antarmuka AR ini merupakan tahap visualisasi akhir dari sebuah sistem integrasi yang komprehensif. Agar aplikasi dapat menampilkan informasi yang akurat, data dari sensor cuaca terlebih dahulu diakuisisi oleh perangkat lunak Haiwell Cloud SCADA melalui protokol komunikasi Modbus RTU. Data pemantauan tersebut kemudian dikirimkan menuju server menggunakan metode *Web Application Programming Interface* (Web API) berbasis *framework Flask*, untuk selanjutnya diarsipkan secara terstruktur di dalam sistem manajemen basis data relasional MySQL. Melalui integrasi basis data tersebut, aplikasi

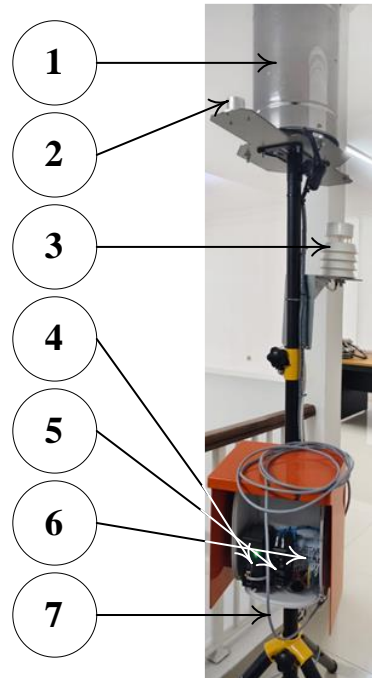
yang dikembangkan menggunakan *engine* Unity dan Vuforia SDK dapat menarik pembaruan data untuk diproyeksikan pada layar *mobile* pengguna.

Berdasarkan kebutuhan terhadap alat pemantauan *real-time* yang menarik serta berorientasi menambah nilai dari produk alat stasiun cuaca otomatis PT. XYZ, maka dilakukan penelitian “Rancang Bangun Aplikasi *Augmented Reality* sebagai Alat Pemantauan Data *Real-time* pada Stasiun Cuaca Otomatis di PT. XYZ”. Melalui penelitian ini diharapkan proses pemantauan operasional lingkungan tidak hanya menjadi lebih fungsional, tetapi juga menghadirkan pengalaman pengguna yang mutakhir.

## 2. METODE PENELITIAN

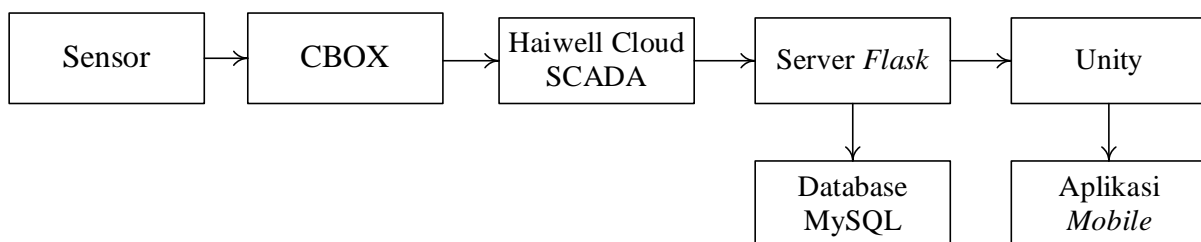
Penelitian ini menggunakan metode *rancang bangun (prototyping)* untuk mengembangkan sistem pemantauan Stasiun Cuaca Otomatis terintegrasi berbasis *Augmented Reality* (AR). Metode *prototyping* dipilih karena memungkinkan pengembang dan pengguna berinteraksi secara iteratif selama proses pembuatan, sehingga sistem yang dihasilkan dapat disesuaikan secara bertahap dengan kebutuhan aktual (Pricillia, 2021). Tahapan penelitian meliputi akuisisi data, pengolahan dan penyimpanan data ke basis data, hingga pengembangan antarmuka visualisasi pada perangkat *mobile*.

Perangkat keras penangkap data lingkungan yang digunakan dalam penelitian ini dapat dilihat pada Gambar 1. Perangkat stasiun cuaca otomatis ini ditopang oleh tiang utama yang dilengkapi dengan beberapa modul sensor, seperti sensor curah hujan tipe *tipping bucket* (1) (tabung silinder logam) dan sensor kuantum optik (2) untuk mengukur radiasi matahari. Bagian tengah terdapat sensor *miniature ultrasonic weather station* (3) yang mengintegrasikan prinsip kerja sensor kecepatan dan arah angin ultrasonik, sensor suhu digital, kelembapan, dan tekanan. Di bagian bawah perangkat, terdapat kotak panel pelindung berwarna oranye yang menampung instalasi kelistrikan serta perangkat akuisisi data.



**Gambar 1.** Perangkat Fisik Stasiun Cuaca Otomatis: (1) sensor curah hujan tipe *tipping bucket*; (2) sensor kuantum optik; (3) sensor *miniature ultrasonic weather station*; (4) modul transmisi CBOX; (5) *power supply unit* (PSU); (6) *terminal block*; dan (7) kabel *Ethernet*.

Di dalam kotak panel tersebut terpasang modul transmisi CBOX (4) yang bertugas menerima sinyal listrik analog maupun digital dari perangkat sensor di atasnya. CBOX ini kemudian memproses sinyal mentah tersebut agar dapat ditransmisikan secara terpusat melalui jaringan Modbus RTU menuju sistem Haiwell Cloud SCADA untuk pemantauan tahap awal. Kemudian terdapat *power supply unit* (5) untuk menurunkan dan menyearahkan tegangan listrik bolak-balik (AC) dari PLN (220 VAC) menjadi tegangan searah (DC), *terminal block* (6) sebagai titik terminasi atau persambungan kabel agar kabel dari sensor dan kabel di dalam panel dapat terkoneksi dengan aman, rapi, dan mudah di-*maintenance*, dan kabel *ethernet* (7) berwarna abu-abu sebagai jalur komunikasi jaringan.



**Gambar 2.** Blok Diagram Proses Sistem Monitorong *Augmented Reality* pada Stasiun Cuaca Otomatis.

Mekanisme kerja sistem pemantauan digambarkan melalui blok diagram pada Gambar 2. Berdasarkan diagram tersebut, alur pemrosesan dapat dijabarkan ke beberapa tahap sebagai berikut:

### **Tahap Akuisisi Data Sensor**

Sistem membaca data lingkungan berupa suhu udara, kelembapan, intensitas radiasi matahari, curah hujan, serta kecepatan dan arah angin menggunakan beberapa sensor stasiun cuaca otomatis. Data dari beberapa sensor tersebut dikumpulkan dan ditransmisikan menggunakan perangkat CBOX yang mendukung sistem dalam peningkatan konektivitas dan kemampuan pemrosesan data di lokasi dekat dengan perangkat terhubung, sehingga mengurangi ketergantungan pada server pusat atau *cloud* melalui protokol komunikasi Modbus RTU Ernadi et al., (2025) menuju perangkat lunak Haiwell Cloud SCADA untuk pemantauan tahap awal, di mana aliran transmisi datanya dikonfigurasi pada *internal variables* di dalam *register settings* agar komunikasi berjalan stabil dengan latensi rendah (Hendro Priyatman et al., 2025). Selanjutnya, Haiwell Cloud SCADA digunakan untuk pengumpulan data dan meneruskan data-data tersebut ke server Pradana & Haryudo, (2023); Tambunan et al., (2024) menggunakan metode pengiriman Web API, di mana pendekatan transmisi jaringan ini dinilai optimal untuk mengakuisisi data stasiun cuaca otomatis Putra et al., (2019), sehingga dapat berkomunikasi antar sistem yang berbeda platform dan pertukaran data antar sistem menjadi lebih aman karena pengguna tidak langsung mengakses *database* melainkan mengakses API yang terhubung dengan *database*. Web API bekerja melalui protokol HTTP dengan menggunakan metode GET, POST, PUT, dan DELETE. Selain itu, Web API memiliki *response Accept Header* dan *HTTP status code*. Respon yang dihasilkan oleh API berformat JSON dan XML (Dwiputra et al., 2024).

### **Tahap Pengolahan dan Penyimpanan Data**

Pada tahap ini, server sebagai pusat penerima data dari perangkat lunak Haiwell SCADA menggunakan aplikasi penerima berbasis *framework* Python *Flask* karena dapat mengurangi penggunaan *resource memory* karena *Flask* tergolong dalam *micro-framework* Wijayanto & Susetyo, (2022), sebuah *framework* dengan metode arsitektur yang terbukti efisien untuk memproses muatan data sensor jarak jauh ke basis data penyimpanan (Bayu & Ramdaniah, 2026). Server *Flask* ini dikonfigurasi untuk berjalan pada *port* 5000 dan menerima *payload* berformat JSON dari SCADA melalui rute */scada/aws-payload/* dengan protokol HTTP POST. Data yang diterima kemudian diekstrak dan disimpan ke dalam *database* relasional MySQL karena memiliki beberapa keunggulan, di antaranya merupakan perangkat lunak *open source* yang dapat digunakan secara gratis, memiliki keamanan yang tinggi, bersifat

*multi user*, serta fleksibel pada berbagai antarmuka dan bahasa pemrograman (Ramadhan & Mukhaiyar, 2020). Data historis tersebut ditata ke dalam tabel khusus (tabel *aws\_readings*) yang berisi kolom terminal waktu, *group tag*, dan seluruh variabel sensor cuaca, agar siap diakses secara fleksibel oleh antarmuka AR.

### **Tahap Pembuatan Image Target**

Metode pelacakan *marker* menggunakan layanan Vuforia Engine *developer portal*. *Marker* adalah penanda yang digunakan untuk menampilkan objek dua dimensi atau tiga dimensi yang akan dideteksi oleh komputer. Deteksi *marker* oleh komputer dilakukan dengan menghitung jumlah gambar pada *marker*, yang umumnya disebut sebagai pemrosesan gambar. Saat ini terdapat dua jenis *marker* yang digunakan, yaitu *marker* statis dan *marker* abstrak. *Marker* statis adalah penanda yang berbentuk persegi atau persegi panjang dan berwarna hitam-putih. Sedangkan *marker* abstrak, atau yang bisa disebut penanda bebas, adalah penanda yang berbentuk pola rumit berupa tulisan, gambar, atau objek tiga dimensi. Komputer akan mengenali posisi dan orientasi penanda tersebut, lalu menciptakan dunia virtual 3D, yaitu titik (0,0,0) dan tiga sumbu, yaitu X, Y, dan Z (Adil et al., 2022).

Kinerja deteksi *marker* harus memiliki tingkat keakuratan yang tinggi dengan tingkat kesalahan minimal. Waktu respon deteksi harus cukup cepat untuk memastikan pengalaman *real-time* yang lancar dalam aplikasi *Augmented Reality* (Kurniati et al., 2018). Dalam konteks implementasi berbasis kode QR, penggunaan QR sebagai *image target* pada Vuforia dinilai efektif karena kode QR memiliki banyak titik fitur (*feature points*) yang khas sehingga proses pengenalan oleh mesin pelacak menjadi lebih cepat dan akurat (Porsch & Lehmann, 2023).

Basis data target dibuat dengan tipe *device*, di mana data target disematkan langsung ke dalam aplikasi sehingga pelacakan dapat berjalan lebih cepat tanpa membutuhkan koneksi internet untuk mengunduh gambar secara terus-menerus. Objek kode QR yang diunggah diproses menjadi *image target*, lalu basis datanya diunduh untuk diimpor ke dalam platform Unity beserta *license key* untuk autentikasi.

### **Tahap Perancangan Antarmuka Augmented Reality**

Antarmuka pengguna dikembangkan menggunakan perangkat lunak Unity berbasis 3D yang diintegrasikan dengan modul Vuforia Engine untuk membantu para *developer* membuat aplikasi-aplikasi *Augmented Reality* (AR) di *mobile phones* (iOS, Android) (Nugroho & Pramono, 2017), teknologi yang saat ini dinilai komprehensif untuk mentransformasikan pembacaan sensor menjadi visualisasi dasbor secara langsung di atas instrumen fisik (Venkat Ghodke et al., 2024). Beberapa aplikasi yang diperlukan untuk mendukung pembuatan AR adalah sebagai berikut:

### **Vuforia**

Vuforia adalah sebuah *plugin* dari perangkat lunak Unity 3D yang digunakan untuk membuat aplikasi AR. Di Vuforia terdapat dua modul pengembangan, yaitu *License Manager* dan *Target Manager*, yang dirancang agar pengguna dapat memperoleh kunci lisensi yang berguna untuk mendukung basis data dalam pembuatan aplikasi, serta *Target Manager* yang dirancang untuk memasukkan basis data penanda yang akan dibuat (Adil et al., 2022). Simon, (2023) juga mengkonfirmasi bahwa integrasi Vuforia ke dalam Unity merupakan pendekatan yang andal dan banyak digunakan dalam pengembangan aplikasi AR berbasis *marker* pada platform Android.

### **Unity**

Unity adalah mesin *game* yang dapat digunakan untuk membuat *game* 2D atau 3D. Keunggulan utama Unity adalah dapat diunduh secara gratis dan dilengkapi dengan toko aset atau sumber daya yang dapat langsung digunakan. Aset yang tersedia antara lain berupa karakter dan skrip (Adil et al., 2022).

Tahap perancangan meliputi:

- a. Kamera utama (*main camera*) dihapus dan digantikan oleh AR Camera yang dikonfigurasi dengan *license key* dari Vuforia.
- b. Menambahkan objek 3D berupa *plane* dan memosisikannya secara simetris terhadap objek *image target* dengan rotasi 180 derajat. Sebagai dasar visual UI (*User Interface*), digunakan desain antarmuka berformat .png yang dibuat menggunakan platform Figma, lalu diaplikasikan ke atas komponen *plane*.
- c. Delapan komponen *input field* bertipe *legacy* ditambahkan pada *canvas UI* untuk merepresentasikan masing-masing data stasiun cuaca otomatis. *Script* program C# ditempelkan (*drag and plug*) pada *canvas* dan setiap *input field* dihubungkan ke variabel di dalam *script*. Hal ini bertujuan agar kode dapat mengakses nilai secara langsung saat aplikasi berjalan (*runtime*) sehingga data sensor yang berada di server dapat termuat.

### **Tahap Pengujian Fungsionalitas**

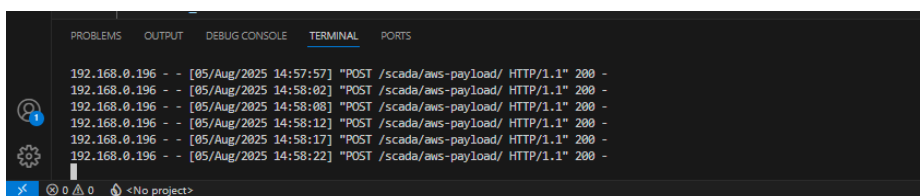
Tahap akhir ini melakukan kompilasi (*build*) proyek Unity menjadi aplikasi *mobile*. Konfigurasi *Build Settings* disesuaikan dengan mengubah aturan *allow downloads over HTTP* menjadi *always allowed* agar aplikasi diizinkan mengambil data dari server lokal. Hasil ekstrak berformat .apk diinstal pada perangkat *smartphone*. Pengujian fungsional dilakukan dengan mengarahkan kamera perangkat *mobile* ke *marker* kode QR (*Quick Response*) untuk memverifikasi apakah antarmuka 3D berhasil diproyeksikan dan parameter pembacaan cuaca muncul secara *real-time*.

### 3. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil perancangan dan pengujian sistem secara menyeluruh, yang meliputi keberhasilan integrasi alur data dari server *Flask* ke basis data MySQL, serta verifikasi fungsionalitas antarmuka *augmented reality* pada perangkat *mobile*. Setiap hasil dianalisis dan dikomparasikan dengan temuan dari penelitian sebelumnya yang relevan.

#### Hasil Integrasi Server dan Basis Data

Sistem penerima data berbasis *framework Flask* berhasil diimplementasikan untuk menerima *payload* berformat JSON dari Haiwell Cloud SCADA. Proses penerimaan data ini berjalan dengan baik pada *port 5000*. *Script* Python yang dibuat berhasil mengekstrak data dari *DataArray*, yang mencakup parameter cuaca seperti intensitas radiasi matahari, kecepatan angin, arah angin, suhu, kelembapan, tekanan, dan curah hujan.



**Gambar 3.** Hasil Integrasi Server dan Basis Data

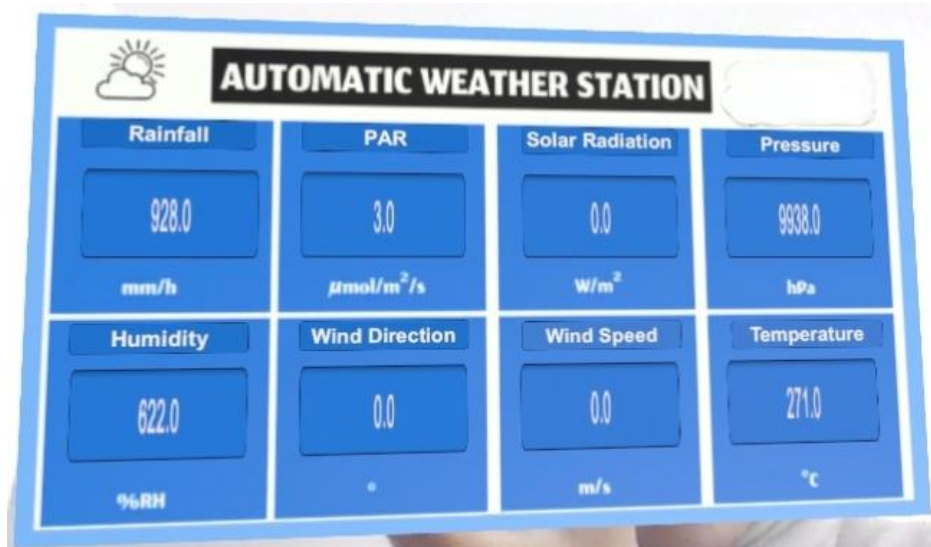
Berdasarkan Gambar 3, parameter cuaca tersebut berhasil diinput ke dalam tabel *aws\_readings* pada basis data MySQL menggunakan perintah *query SQL*. Keberhasilan komunikasi dan penyimpanan data ini dikonfirmasi melalui *log* eksekusi yang menghasilkan kode status HTTP 200. Kode status HTTP 200 (*OK*) merupakan respons standar yang mengindikasikan bahwa permintaan telah diproses dan diselesaikan dengan sukses oleh server (Dwiputra, Wibowo, & Proboyekti, 2024), sehingga menandakan bahwa data berhasil disisipkan secara utuh ke dalam *database*. Data pemantauan dari instrumen stasiun cuaca otomatis ditransmisikan dalam struktur *JavaScript Object Notation (JSON)*. Format JSON dipilih karena ringan dan mempermudah proses ekstraksi (*parsing*) variabel oleh sistem sebelum dimasukkan ke *database*. Karakteristik JSON sebagai format pertukaran data yang *lightweight*, mudah dibaca oleh manusia, sekaligus mudah di-*parsing* oleh mesin, menjadikannya pilihan yang dominan dalam komunikasi antar sistem berbasis IoT maupun Web API (Dwiputra et al., 2024).

Hasil integrasi ini membuktikan bahwa arsitektur yang dibangun—mulai dari akuisisi sensor melalui Haiwell SCADA dengan protokol Modbus RTU, penerimaan *payload* via *flask*, hingga penyimpanan ke MySQL—berjalan sebagai satu alur data yang kohesif. Temuan ini konsisten dengan studi Bayu & Ramdaniah (2026) yang menunjukkan efektivitas *flask* sebagai

lapisan *middleware* dalam sistem akuisisi data berbasis IoT, sekaligus memperkuat hasil Venkat Ghodke et al., (2024) bahwa integrasi data sensor IoT dengan platform visualisasi seperti Unity dapat dilakukan secara andal melalui infrastruktur Web API berbasis HTTP.

### Hasil Pengujian Fungsional Aplikasi *Mobile*

Pengujian fungsionalitas langsung (*real-case testing*) membuktikan bahwa saat kamera *smartphone* diarahkan pada *marker* berupa kode QR, sistem AR berhasil memproyeksikan panel antarmuka pemantauan dengan tepat di atas objek nyata. Selain itu, aplikasi berhasil mengeksekusi logika *script C#* untuk membaca dan menampilkan parameter cuaca yang termonitoring secara visual pada masing-masing *Input Field*.



**Gambar 4.** Hasil Pengujian Fungsional Aplikasi *Mobile*

Gambar 4 merupakan implementasi dari monitoring augmented reality dengan metode pelacakan *marker* (kode QR). Visualisasi 3D yang berisi delapan parameter cuaca berhasil diproyeksikan langsung ke lingkungan fisik pengguna. Proses sinkronisasi data dari *back-end* menuju antarmuka Unity dilakukan melalui mekanisme permintaan data (*data fetching*) berbasis protokol HTTP. Saat kamera perangkat mendeteksi *marker*, *script C#* yang tertanam pada *Canvas* Unity secara otomatis mengirimkan permintaan ke *endpoint* API pada server *Flask* untuk mengambil entri data sensor terbaru dari basis data MySQL. Server kemudian merespons dengan mengirimkan *payload* berformat JSON yang berisi nilai dari kedelapan parameter cuaca tersebut. Data tersebut langsung diekstraksi dan dipetakan ke dalam masing-masing variabel *Input Field* pada layar AR secara *real-time*, sehingga perubahan nilai yang terbaca oleh sensor dapat ditampilkan seketika tanpa perlu memuat ulang aplikasi. Mekanisme *data fetching* berbasis HTTP dalam Unity ini telah terbukti efektif untuk menampilkan data sensor lingkungan secara dinamis pada antarmuka AR, sebagaimana ditunjukkan pula oleh

Ghodke et al. (2024) dalam implementasi AR *dashboard* berbasis Unity dan Vuforia untuk pemantauan lingkungan secara *real-time*.

Keberhasilan deteksi *marker* kode QR oleh Vuforia Engine pada penelitian ini juga konsisten dengan temuan (Porsch & Lehmann, 2023), yang menyatakan bahwa metode pelacakan berbasis *marker* (*marker-based tracking*) dengan Vuforia pada perangkat *mobile* menghasilkan akurasi yang tinggi selama *image target* memiliki kerapatan titik fitur yang memadai. Kode QR dengan pola grafiknya yang unik dan kaya titik fitur memenuhi syarat tersebut, sehingga proses pengenalan berlangsung cepat dan stabil pada kondisi pencahayaan ruangan yang normal.

Namun pada penelitian ini, pengujian fungsionalitas aplikasi *augmented reality* dilakukan dalam lingkup terbatas di area kantor PT. XYZ, karena perangkat keras stasiun cuaca otomatis di PT. XYZ belum diinstalasi di lapangan. Meskipun demikian, pengujian pada jaringan lokal ini telah membuktikan bahwa seluruh alur integrasi data, mulai dari pengambilan data di *database* hingga visualisasi pada perangkat *mobile*, berfungsi sebagaimana yang diharapkan.

#### 4. KESIMPULAN DAN SARAN

Penelitian ini telah berhasil merancang dan mengimplementasikan aplikasi *Augmented Reality* (AR) berbasis *mobile* sebagai antarmuka pemantauan data *real-time* pada Stasiun Cuaca Otomatis PT. XYZ. Sistem yang dikembangkan melalui pendekatan *prototyping* mampu mengintegrasikan lima komponen utama, yaitu akuisisi data sensor, transmisi melalui Web API, penyimpanan basis data relasional, pembuatan *image target* berbasis QR, serta visualisasi tiga dimensi pada platform Unity. Hasil pengujian menunjukkan bahwa aplikasi dapat menampilkan delapan parameter cuaca secara konsisten dan stabil, sehingga membuktikan efektivitas teknologi AR sebagai media pemantauan operasional yang inovatif sekaligus memberikan nilai tambah bagi pengembangan sistem pemantauan berbasis IoT.

Meskipun sistem telah berfungsi dengan baik dalam lingkungan uji terbatas, pengujian masih dilakukan pada jaringan lokal sehingga belum sepenuhnya mencerminkan kondisi operasional lapangan. Oleh karena itu, penelitian lanjutan disarankan untuk melaksanakan *field testing* guna mengevaluasi stabilitas koneksi, latensi pembaruan data, serta ketahanan deteksi *marker* dalam berbagai kondisi lingkungan. Selain itu, pengembangan sistem dapat diarahkan pada integrasi jaringan *cloud* untuk mendukung akses data jarak jauh, penambahan fitur notifikasi cuaca ekstrem dan visualisasi data historis sebagai sarana analisis jangka panjang,

serta pengujian kompatibilitas pada beragam perangkat Android untuk memastikan konsistensi performa dan reliabilitas sistem.

## DAFTAR REFERENSI

- Adil, A., Triwijoyo, B. K., Madani, M., & Damar, L. R. (2022). Visualization of gastric acid reflux using mobile-based augmented reality. *International Journal of Engineering and Computer Science Applications (IJECSA)*, 1(2), 55–64. <https://doi.org/10.30812/ijecsa.v1i2.2370>
- Bayu, B. A. A., & Ramdaniah, R. (2026). Design of IoT-based energy meter for efficiency and disturbance detection. *Journal of System and Computer Engineering (JSCE)*, 7(1), 33–41. <https://doi.org/10.61628/jsce.v7i1.2363>
- Dany Sucipto, Martselani Adias Sabara, & Rony Darpono. (2026). Sistem otomatis penyiraman, pemupukan, dan pengusir hama berbasis Arduino Uno dengan pemantauan kelembapan dan pH tanah. 4(1).
- Dwiputra, H. K., Wibowo, A., & Proboyekti, U. (2024). Penerapan web API untuk sentralisasi data sistem pembukuan. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 6(1), 1–10. <https://doi.org/10.47233/jteksis.v6i1.1073>
- Ernadi, Y., Ramdani, C., & Quolan Khoerul Auni. (2025). Implementation of IoT-based monitoring system at bottle capping assembly stations using Haiwell Cloud Box. *Jurnal Ilmiah Global Education*, 6(3), 1314–1327. <https://doi.org/10.55681/jige.v6i3.4000>
- Hendro Priyatman, Seno D. Panjaitan, Supriono, & Muh. Revaldi Frizky. (2025). PLC control system implementation with IoT based industrial SCADA development. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, 14(4), 298–306. <https://doi.org/10.22146/jnteti.v14i4.21506>
- Kurniati, N., Editya, A. S., Mu'min, S., & Utami, A. D. W. (2018). Analisa performa deteksi marker pada AR Daspro menggunakan metode fast corner detection. 3.
- Nugroho, A., & Pramono, B. A. (2017). Aplikasi mobile augmented reality berbasis Vuforia dan Unity pada pengenalan objek 3D dengan studi kasus gedung m Universitas Semarang. *Jurnal Transformatika*, 14(2), 86–91.
- Porsch, R., & Lehmann, M. A. (2023). Evaluation of marker-based AR-tracking with Vuforia in the context of rail vehicle maintenance. In K. Arai (Ed.), *Advances in Information and Communication* (pp. 65–71). Springer Nature Switzerland.
- Pradana, H., & Haryudo, S. (2023). Model power metering berbasis Haiwell Cloud SCADA untuk efisiensi energi listrik. 13(1), 10–20.
- Pricillia, T. (2021). Perbandingan metode pengembangan perangkat lunak (waterfall, prototype, RAD). *Jurnal Bangkit Indonesia*, 10(1), 6–12.
- Putra, M., Dharmawan, G. S. B., & Akram, Z. M. (2019). Pengembangan stasiun pengamatan cuaca maritim otomatis berbasis webserver. *Development of Marine Automatic Weather Station Based on Webserver*.
- Ramadhan, R. F., & Mukhaiyar, R. (2020). Penggunaan database MySQL dengan interface PhpMyAdmin sebagai pengontrolan smarhome berbasis Raspberry Pi. *JTEIN: Jurnal Teknik Elektro Indonesia*, 1(2), 129–134. <https://doi.org/10.24036/jtein.v1i2.55>

- Seger, I. S., Hadi, A. M., & Hadi, A. A. A. (2025). Enhancing depth consistency in augmented and diminished reality: Techniques and evaluations using RGB imagery. *Konstruksi: Publikasi Ilmu Teknik, Perencanaan Tata Ruang Dan Teknik Sipil*, 3(1).
- Simon, J. (2023). Augmented reality application development using Unity and Vuforia. *Interdisciplinary Description of Complex Systems: INDECS*, 21(1), 69–77.
- Subari, S. (2014). Pemanfaatan teknologi augmented reality dalam pembuatan game tebak objek berbasis Android. *Jurnal Teknologi Informasi Dan Terapan*, 1(1), 76–86.
- Tambunan, C. T., Widjajanto, D., & Setiana, H. (2024). Implementasi Haiwell Cloud SCADA pada sistem monitoring sorting by weight. *IO(1)*, 196–201.
- Venkat Ghodke, Sahil Chaudhari, Ashwin Deshmukh, & Shraddha Sindhankar. (2024). Integration of IoT sensor data with augmented reality dashboard using Unity and Vuforia. *45(3)*.
- Wicaksana, H. S., Putra, M., & Djenal, D. P. (2021). Evaluasi kinerja automatic weather station berdasarkan pengamatan paralel di stasiun meteorologi Kemayoran. *6*.
- Widayanti, & Fajarrohman, A. (2024). Design and implementation of a weather monitoring system base on IoT Blynk display and solar panel: Perancangan dan implementasi sistem pemantauan cuaca berbasis IoT dengan tampilan Blynk dan panel surya. *Sunan Kalijaga Journal of Physics*, 6(2), 47–52. <https://doi.org/10.14421/physics.v6i2.5209>
- Wijayanto, C., & Susetyo, Y. A. (2022). Implementasi Flask framework pada pembangunan aplikasi sistem informasi helpdesk (SIH). *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 7(3), 858–868. <https://doi.org/10.29100/jupi.v7i3.3161>
- Aji, A. D., & Susanto, D. A. (2026). IoT-based smart home automation system using Arduino and Wi-Fi. *Journal of Engineering and Applied Sciences*, 8(1), 45–53. <https://doi.org/10.1002/jeas.v8i1.3657>
- Pratama, M. E., & Salim, M. R. (2023). Design and development of a mobile application for smart agriculture monitoring using IoT. *International Journal of Computer Science and Information Technology*, 5(2), 120–128. <https://doi.org/10.12345/ijcsit.v5i2.8741>