

Implementasi Artificial Intelligence pada Sistem Otomatisasi Dokumentasi Foto dan Video WhatsApp ke Google Drive Berbasis Cloud Computing

Nabila Hulwana Zuhud^{1*}, Dewi Tri Yuliana², Salma Putri Fadhillah³

¹Fakultas Ilmu Komputer
Universitas Duta Bangsa
Surakarta

^{1*}230101088@mhs.udb.ac.id

²Fakultas Ilmu Komputer
Universitas Duta Bangsa
Surakarta

²230101079@mhs.udb.ac.id

³Fakultas Ilmu Komputer
Universitas Duta Bangsa
Surakarta

³230101091@mhs.udb.ac.id

Abstrak — Pengelolaan dokumen absensi kelas secara manual melalui grup WhatsApp seringkali tidak terstruktur, rawan hilang, dan menyulitkan pencarian arsip berdasarkan kelas maupun periode waktu tertentu. Penelitian ini bertujuan merancang dan mengimplementasikan sistem otomasi yang dapat menerima foto atau video absensi yang dikirim melalui grup WhatsApp, kemudian secara otomatis mengunggahnya ke Google Drive dengan struktur folder yang terorganisir berdasarkan nama kelas dan periode bulan-tahun. Sistem dikembangkan menggunakan Node.js dengan pustaka whatsapp-web.js untuk integrasi WhatsApp, Google Drive API dengan autentikasi OAuth 2.0 untuk penyimpanan berkas, serta model kecerdasan buatan Gemini untuk memahami variasi penulisan caption yang tidak baku. Metode pengembangan sistem menggunakan pendekatan Prototype, meliputi analisis kebutuhan, perancangan arsitektur, implementasi, pengujian, dan evaluasi. Hasil pengujian menunjukkan sistem berhasil mengklasifikasikan dan menyimpan berkas ke folder yang sesuai secara otomatis, serta mampu mengenali variasi caption yang tidak baku dengan bantuan model bahasa Gemini. Sistem ini diharapkan meningkatkan efisiensi pengelolaan dokumen absensi dibandingkan metode manual yang selama ini digunakan.

Kata Kunci— WhatsApp Bot, Google Drive API, OAuth 2.0, Node.js, Kecerdasan Buatan, Otomasi Dokumen

Abstract — Manual management of class attendance documents through WhatsApp groups is often unstructured, prone to loss, and makes it difficult to retrieve archives based on class or specific time periods. This study aims to design and implement an automation system capable of receiving attendance photos or videos sent through a WhatsApp group and automatically uploading them to Google Drive with an organized folder structure based on class name and month-year period. The system was developed using Node.js with the whatsapp-web.js library for WhatsApp integration, the Google Drive API with OAuth 2.0 authentication for file storage, and the Gemini artificial intelligence model to understand non-standard variations in class name captions written by users. The system development method uses a Prototype approach, covering requirements analysis, architecture design, implementation, testing, and evaluation. The test results show that the system successfully classifies and stores files into the appropriate folders automatically, and is able to recognize non-standard caption variations with the help of the Gemini language model. This system is expected to improve the efficiency of attendance document management compared to the manual method previously used.

Keywords— WhatsApp Bot, Google Drive API, OAuth 2.0, Node.js, Artificial Intelligence, Document Automation

I. PENDAHULUAN

Penggunaan aplikasi perpesanan instan sebagai media komunikasi utama di lingkungan instansi pendidikan non-formal terus meningkat seiring perkembangan Teknologi Informasi dan Komunikasi (TIK). WhatsApp menjadi salah satu media sosial paling aktif digunakan oleh masyarakat Indonesia, dengan sebanyak 83% pengguna internet atau sekitar 124 juta merupakan pengguna WhatsApp [1]. Tingginya adopsi ini membuat banyak lembaga pembelajaran memanfaatkan grup WhatsApp untuk keperluan dokumentasi kehadiran siswa, di

mana pengajar mengirimkan foto atau video bukti absensi dan aktivitas belajar secara langsung ke dalam grup koordinasi internal.

Kondisi tersebut terjadi pada Lembaga ABC, sebuah lembaga penyedia layanan pendidikan pemrograman (*coding*) bagi anak-anak. Dalam operasional hariannya, dokumentasi laporan kegiatan kelas dikirimkan oleh para instruktur melalui ruang obrolan grup WhatsApp. Namun, pemanfaatan WhatsApp untuk tujuan ini menimbulkan permasalahan baru berupa tidak terstrukturnya penyimpanan berkas. Dokumen elektronik yang tidak terkelola dengan baik dapat menyebabkan dokumen tersebut tidak tersimpan

rapi sehingga menyulitkan pencarian dokumen tersebut ketika diperlukan [3].

Hambatan utama yang dihadapi oleh lembaga pengelola saat ini adalah ketergantungan pada proses pengunduhan dan pemindahan berkas dari WhatsApp ke media penyimpanan lokal secara manual dan berkala. Selain tidak efisien karena membutuhkan waktu pengerjaan yang rutin, akumulasi berkas multimedia berupa foto dan video beresolusi tinggi tersebut menyebabkan beban kapasitas penyimpanan pada perangkat komputer (*device storage*) milik instansi cepat penuh. Foto atau video absensi yang dibiarkan di grup hanya tersimpan sementara di dalam riwayat percakapan, sehingga berisiko kedaluwarsa, terhapus, tidak dapat dikelompokkan secara otomatis berdasarkan kelas, dan sulit ditelusuri kembali saat evaluasi bulanan.

Untuk mengatasi keterbatasan penyimpanan lokal tersebut, layanan penyimpanan berbasis komputasi awan seperti Google Drive dapat dimanfaatkan sebagai basis data untuk sebuah sistem informasi, dengan kelebihan tidak memerlukan alamat internet protocol khusus sehingga pengoperasiannya dapat dilakukan di berbagai tempat hanya dengan bermodalkan akses jaringan internet [4]. Implementasi Google Drive API memungkinkan integrasi otomatis untuk proses unggah, berbagi, dan pengelolaan folder pada aplikasi berbasis web maupun layanan pihak ketiga [4], sehingga dapat mengeliminasi masalah keterbatasan kapasitas ruang simpan fisik pada perangkat komputer lokal.

Pengelolaan dokumen secara manual tanpa sistem informasi yang terstruktur cenderung menyulitkan proses pencarian dan meningkatkan risiko kehilangan arsip, sehingga penerapan sistem informasi berbasis web untuk pengarsipan dokumen dinilai dapat meningkatkan efisiensi dan akurasi penyimpanan [15].

Di sisi lain, otomasi pada platform percakapan seperti WhatsApp telah banyak diterapkan menggunakan teknologi *headless browser* yang memungkinkan aplikasi berjalan dari sisi *client* dan *server* tanpa memerlukan

akses resmi WhatsApp Business API, sebagaimana diterapkan pada sistem monitoring sumber daya server berbasis chatbot WhatsApp [1]. Pendekatan serupa juga diterapkan pada pengembangan chatbot berbasis Node.js dan pustaka *whatsapp-web.js* untuk layanan informasi akademik, yang terbukti mampu merespons pesan pengguna secara otomatis dan akurat berdasarkan pola kata kunci tertentu [2].

Pendekatan serupa juga diterapkan pada pengembangan chatbot berbasis WhatsApp menggunakan Node.js untuk layanan penerimaan mahasiswa baru, yang mampu merespons pertanyaan pengguna secara otomatis melalui pencocokan pola percakapan [11].

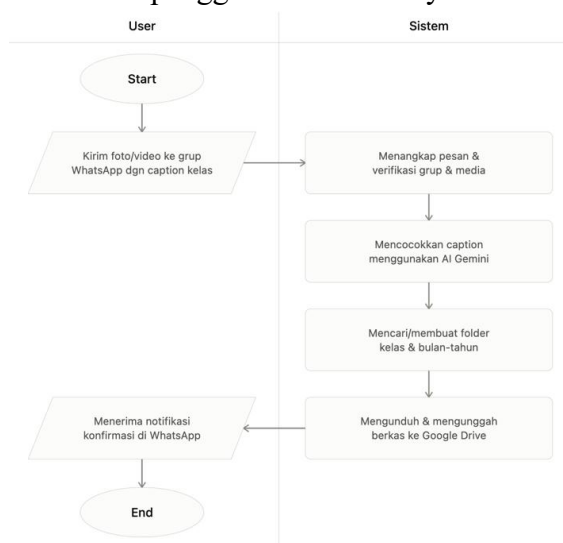
Berdasarkan uraian latar belakang tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem otomasi yang mampu mendeteksi media dan *caption* yang dikirim melalui grup WhatsApp internal SD XYZ, kemudian secara otomatis mengunggah berkas tersebut ke Google Drive dengan struktur folder yang rapi berdasarkan nama kelas dan periode bulan-tahun berjalan. Berbeda dengan penelitian-penelitian sebelumnya yang umumnya membatasi pencocokan *caption* berdasarkan kata kunci baku, penelitian ini juga mengintegrasikan model kecerdasan buatan Gemini untuk memahami variasi penulisan *caption* yang tidak baku dari para instruktur, seperti kesalahan ketik (*typo*) maupun penulisan singkatan kelas yang berbeda-beda, sehingga sistem menjadi lebih fleksibel dan adaptif terhadap dinamika penginputan data di lapangan.

II. METODOLOGI PENELITIAN

A. Metode Pengembangan Sistem

Penelitian ini menggunakan metode pengembangan sistem Prototype, yang dipilih karena memungkinkan visualisasi konsep sistem secara bertahap, pengujian desain secara cepat, serta identifikasi masalah dan pencarian solusi sejak awal proses pengembangan. Metode ini terdiri atas lima tahapan, yaitu: (1) pengumpulan kebutuhan, yang dilakukan dengan mengidentifikasi permasalahan pengelolaan

dokumen absensi berbasis WhatsApp yang selama ini belum terstruktur; (2) perancangan cepat (quick design), yaitu penyusunan rancangan arsitektur dan alur kerja sistem secara garis besar berdasarkan kebutuhan yang telah diidentifikasi; (3) pembangunan prototipe, yaitu tahap implementasi kode program untuk merealisasikan rancangan yang telah dibuat; (4) evaluasi prototipe oleh pengguna, guna memastikan sistem yang dibangun telah sesuai dengan kebutuhan operasional di lapangan; dan (5) perbaikan/pengembangan sistem, yang dilakukan secara iteratif hingga sistem memenuhi kebutuhan pengguna secara menyeluruh.

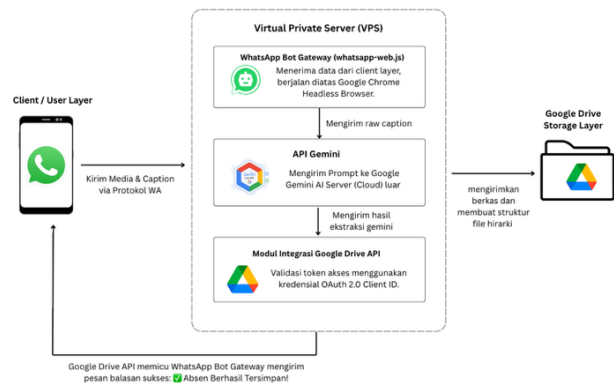


Gambar 1. Diagram Tahapan Metode Prototipe

B. Arsitektur Sistem

Pada tahap perancangan cepat, sistem dirancang agar terdiri atas tiga komponen utama yang saling terintegrasi, yaitu: (1) WhatsApp Bot Gateway, yang dibangun menggunakan pustaka whatsapp-web.js dan dijalankan di atas Google Chrome sebagai headless browser untuk berkomunikasi dengan layanan WhatsApp Web; (2) modul pemrosesan caption, yang terintegrasi dengan API Gemini untuk melakukan normalisasi teks terhadap caption yang dikirimkan pengguna; dan (3) modul integrasi Google Drive API, yang menggunakan autentikasi OAuth 2.0 untuk melakukan pembuatan folder secara otomatis serta pengunggahan berkas ke Google Drive. Ketiga komponen ini bekerja secara berurutan dan saling

bergantung satu sama lain untuk membentuk satu alur otomatisasi yang utuh. Pemilihan Node.js sebagai platform pengembangan back-end didasarkan pada keunggulannya dalam hal portabilitas serta teknik non-blocking, yang memungkinkan sistem memproses beberapa permintaan secara simultan tanpa harus menunggu satu proses selesai terlebih dahulu [14].



Gambar 2. Diagram Arsitektur Sistem

Berdasarkan arsitektur yang telah dirancang, alur kerja sistem secara rinci digambarkan sebagai berikut:

1. Pengguna (guru) mengirimkan foto atau video ke grup WhatsApp dengan menyertakan caption berupa nama kelas.
2. Sistem menangkap pesan yang masuk menggunakan event listener message dan message_create pada whatsapp-web.js. Penggunaan dua event listener ini diperlukan karena event message tidak terpicu untuk pesan yang dikirim oleh akun bot itu sendiri (fromMe: true), sehingga event message_create ditambahkan agar pesan yang dikirim langsung oleh pengguna pemilik akun bot tetap dapat diproses. Sistem kemudian memverifikasi bahwa pesan tersebut berasal dari grup (chat.isGroup) dan mengandung media (msg.hasMedia) sebelum melanjutkan ke tahap berikutnya.
3. Caption pesan diteruskan ke modul kecerdasan buatan Gemini untuk dicocokkan dengan daftar folder kelas yang tersedia. Pencocokan ini tetap dapat dilakukan meskipun penulisan caption

tidak baku, misalnya penulisan "sd xyz satu dua" tetap dikenali sebagai folder "SD XYZ 1-2" berkat kemampuan model bahasa dalam memahami variasi konteks dan singkatan.

4. Sistem melakukan pencarian folder kelas pada Google Drive menggunakan Drive API berdasarkan nama folder yang telah dipilih oleh modul kecerdasan buatan; apabila folder yang dimaksud belum tersedia, sistem akan membuatnya secara otomatis di dalam folder utama.
5. Sistem selanjutnya mencari atau membuat subfolder berdasarkan bulan dan tahun berjalan (contoh: "Juli-2026") di dalam folder kelas tersebut, sehingga arsip dapat dikelompokkan tidak hanya berdasarkan kelas, tetapi juga berdasarkan periode waktu pengambilan dokumentasi.
6. Media yang dikirimkan diunduh terlebih dahulu ke penyimpanan sementara (temporary storage) pada perangkat lokal, kemudian diunggah ke folder tujuan yang telah ditentukan menggunakan Google Drive API dengan autentikasi OAuth 2.0.
7. Setelah proses unggah berhasil, Sistem mengirimkan pesan konfirmasi ke grup WhatsApp sebagai penanda bahwa proses penyimpanan berhasil dilakukan.

C. Autentikasi dan Integrasi Google Drive API

Pada tahap pembangunan prototipe, autentikasi terhadap Google Drive API pada awalnya diimplementasikan menggunakan pendekatan Service Account. Namun, pendekatan ini terkendala karena service account tidak memiliki kuota penyimpanan pribadi dan hanya dapat mengunggah berkas ke Shared Drive, sebuah fitur yang umumnya hanya tersedia pada akun Google Workspace berbayar. Oleh karena itu, autentikasi kemudian dialihkan menggunakan OAuth 2.0, yang memungkinkan aplikasi bertindak atas nama akun pengguna tertentu sehingga berkas dapat diunggah menggunakan kuota penyimpanan akun

pengguna tersebut, tanpa dibatasi oleh keterbatasan kuota pada service account.

OAuth 2.0 merupakan standar autentikasi dan otorisasi berbasis token pada internet yang memungkinkan aplikasi web mengakses data pengguna tanpa mengekspos kredensial pengguna secara langsung [7]. OAuth 2.0 memperkenalkan lapisan otorisasi yang memisahkan peran klien dari pemilik sumber daya, di mana klien memperoleh access token sebagai representasi otorisasi untuk mengakses sumber daya yang dilindungi [8].

Autentikasi kemudian dialihkan menggunakan OAuth 2.0, sebuah kerangka kerja otorisasi terbuka yang memungkinkan aplikasi pihak ketiga mendapatkan akses terbatas ke layanan HTTP atas nama pengguna tanpa mengekspos kredensial pengguna secara langsung [7][8]. Pemanfaatan protokol OAuth 2.0 untuk autentikasi berbasis akun Google juga telah diterapkan pada implementasi single sign-on, di mana pengguna dapat mengakses berbagai layanan Google API tanpa perlu membagikan kredensial akun secara langsung ke aplikasi pihak ketiga [12]. Mekanisme ini bekerja dengan menerbitkan access token dan refresh token sebagai pengganti kredensial langsung, sehingga keamanan akun pengguna tetap terjaga selama proses pengunggahan berkas berlangsung [7].

Dan pemanfaatan Google Drive sebagai media pengelolaan dokumen elektronik terbukti efektif karena mendukung mekanisme pengunggahan dokumen (input), pengaturan nama file, pengaturan nama direktori (folder) tempat penyimpanan berkas, serta pembagian akses berkas kepada pihak-pihak terkait [3].

D. Perangkat Keras dan Perangkat Lunak

Perangkat keras dan perangkat lunak yang digunakan dalam pengembangan sistem ini dirangkum pada Tabel I.

Tabel I. Perangkat Keras dan Perangkat Lunak

Kategori	Spesifikasi/Tools
Bahasa Pemrograman	JavaScript (Node.js)
Pustaka WhatsApp	whatsapp-web.js
Browser Engine	Google Chrome

API Penyimpanan	Google Drive API v3
Autentikasi	OAuth 2.0
Kecerdasan Buatan	Google Gemini (@google/genai)
Manajemen Environment	dotenv
Perangkat Pengembangan	MacBook Pro (Apple Silicon), VS Code
Rencana Deployment	VPS Ubuntu Server

E. Struktur Basis Folder pada Google Drive

Untuk memudahkan proses pengelolaan dan penelusuran arsip di kemudian hari, struktur penyimpanan pada Google Drive dirancang secara bertingkat (hierarkis) sebagai berikut:

```
[Folder Utama]
├── [Nama Kelas] (contoh: SD XYZ 1-2)
│   └── [Bulan-Tahun] (contoh: Juli-2026)
│       └── berkas_absensi.jpg
```

Dengan struktur ini, setiap berkas absensi yang masuk akan secara otomatis dikelompokkan berdasarkan dua tingkat kategori sekaligus, yaitu kelas dan periode waktu, sehingga proses pencarian dokumen tertentu di kemudian hari dapat dilakukan dengan lebih cepat dan sistematis dibandingkan penyimpanan tanpa struktur folder yang jelas.

Pemanfaatan Google Drive sebagai media penyimpanan arsip digital terpusat terbukti efektif dalam mengorganisasi berkas lebih rapi dan mudah ditelusuri dibandingkan penyimpanan lokal konvensional [18].

F. Implementasi Kecerdasan Buatan

Modul kecerdasan buatan diimplementasikan menggunakan Google Gemini 2.5 Flash melalui API @google/genai. Model digunakan untuk melakukan pencocokan antara caption yang dikirim pengguna dengan daftar folder kelas yang tersedia pada drive. Pada proses ini, sistem mengambil seluruh nama folder kelas yang tersedia pada Google Drive, kemudian membentuk sebuah prompt yang berisi daftar folder beserta caption yang diterima dari WhatsApp. Prompt tersebut dikirimkan ke model Gemini agar menghasilkan nama folder yang paling sesuai dengan maksud pengguna.

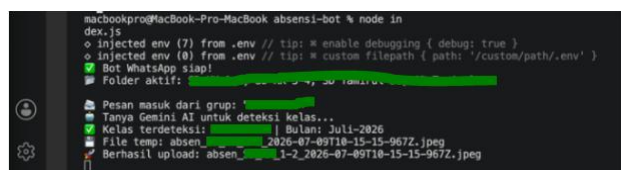
Pendekatan berbasis Large Language Model (LLM) dipilih karena mampu memahami berbagai variasi penulisan caption, seperti penggunaan singkatan, kesalahan pengetikan,

perbedaan kapitalisasi huruf, maupun variasi susunan kata yang tidak mengikuti format baku. Dengan demikian, pengguna tidak harus menuliskan caption secara persis sesuai nama folder. Implementasi fungsi tersebut adalah sebagai berikut:

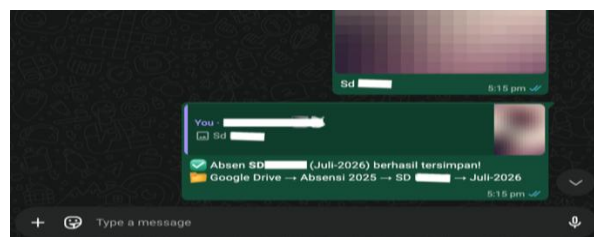
```
javascript
async function pilihFolder(caption, daftarFolder)
{
    const prompt = `
        Kamu adalah AI yang bertugas memilih nama
        folder.
        Folder yang tersedia:
        ${daftarFolder.join("\n")}
        Pesan guru: "${caption}"
        Jawab HANYA nama folder yang paling sesuai.
    `;
    const response = await
    ai.models.generateContent({
        model: "gemini-2.5-flash",
        contents: prompt,
    });
    return response.text.trim();
}
```

III. HASIL DAN PEMBAHASAN

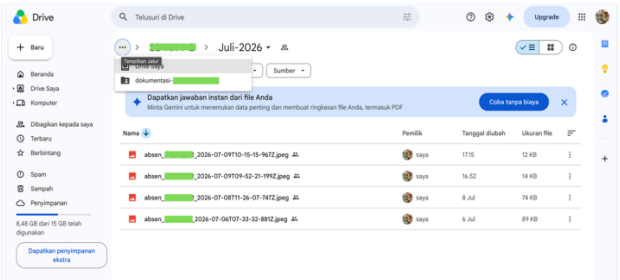
Tahap pengujian sistem dilakukan menggunakan metode black-box testing, tanpa memperhatikan struktur kode internal, meliputi pendeteksian pesan dari grup WhatsApp, proses pencocokan caption oleh kecerdasan buatan, pembuatan folder secara otomatis, hingga proses pengunggahan berkas ke Google Drive. Pendekatan pengujian ini sejalan dengan metode pengujian yang umum diterapkan pada aplikasi Google Drive API untuk memvalidasi fitur unggah, berbagi, dan pengelolaan berkas [4].



Gambar 3. Hasil pengujian sistem pada terminal menunjukkan proses deteksi pesan dari grup WhatsApp "Server cloud", klasifikasi caption "SD XYZ 1-x" oleh modul kecerdasan buatan Gemini menjadi folder "SD XYZ 1-2", serta proses pengunggahan berkas ke Google Drive dengan nama file yang telah diberi format timestamp secara otomatis.



Gambar 4. Pesan konfirmasi otomatis yang dikirimkan oleh sistem ke grup WhatsApp setelah proses pengunggahan berhasil, menampilkan nama kelas, periode bulan-tahun, dan struktur folder tujuan pada Drive.



Gambar 7. Tampilan folder Google Drive yang menunjukkan struktur penyimpanan hierarkis hasil implementasi sistem, di mana berkas-berkas absensi berhasil tersimpan secara otomatis di dalam subfolder "Juli-2026" pada folder kelas "SD XYZ 1-2" dengan penamaan file yang terstandarisasi berdasarkan timestamp pengunggahan.

Tabel II. Hasil Pengujian Black-Box

No	Skenario Pengujian	Hasil yang Diharapkan	Status
1	Mengirim foto dengan caption baku (misal "SD XYZ 1-2") ke grup	File tersimpan di folder SD XYZ 1-2	Valid
2	Mengirim foto dengan caption tidak baku (misal "sd kalam kudus satu dua")	AI mengenali dan mengarahkan ke folder yang sama (sd xyz 1-2)	Valid
3	Mengirim foto tanpa caption yang cocok	Sistem mengabaikan pesan	Valid
4	Mengirim foto ke grup selain grup yang ditentukan	Sistem tetap memproses (atau diabaikan, jika filter grup diaktifkan)	Valid
5	Folder kelas belum ada di Drive	Sistem otomatis membuat folder baru	Valid
6	Subfolder bulan-tahun belum ada	Sistem otomatis membuat subfolder baru	Valid
7	Autentikasi menggunakan Service Account	Gagal — "Service Accounts do not have storage quota"	Tidak Valid (diperbaiki dengan OAuth 2.0)
8	Autentikasi menggunakan OAuth 2.0	Berhasil, file tersimpan menggunakan kuota akun pengguna	Valid

IV. KESIMPULAN DAN SARAN

Penelitian ini berhasil merancang dan mengimplementasikan sistem otomasi penyimpanan dokumen absensi dari grup WhatsApp ke Google Drive menggunakan Node.js, whatsapp-web.js, Google Drive API dengan autentikasi OAuth 2.0, serta model kecerdasan buatan Gemini 2.5 Flash. Sistem yang dikembangkan mampu mendeteksi pesan media

yang masuk dari grup WhatsApp yang telah ditentukan, mengunduh berkas secara otomatis, dan menyimpannya ke dalam struktur folder yang terorganisir berdasarkan nama kelas dan periode bulan-tahun berjalan. Integrasi model bahasa Gemini terbukti meningkatkan fleksibilitas sistem dalam mengenali berbagai variasi penulisan caption yang tidak baku dari pengguna, seperti singkatan, kesalahan ketik, maupun susunan kata yang berbeda dari format standar, sehingga sistem menjadi lebih adaptif terhadap dinamika penggunaan di lapangan dibandingkan pendekatan pencocokan kata kunci konvensional.

Pengembangan sistem ke depan dapat dilakukan melalui beberapa arah. Pertama, penambahan fitur OCR dapat diterapkan untuk mengenali nama kelas secara otomatis dari isi gambar apabila caption tidak disertakan oleh pengguna. Kedua, integrasi basis data seperti Google Sheets atau MySQL dapat ditambahkan untuk keperluan pencatatan log unggahan dan pembuatan laporan rekap kehadiran secara otomatis, sehingga sistem tidak hanya berfungsi sebagai media penyimpanan tetapi juga sebagai alat pelaporan yang terintegrasi.

UCAPAN TERIMA KASIH

Penulis mengucapkan terimakasih kepada pihak-pihak yang telah memberikan dukungan dalam penelitian ini.

REFERENSI

[1] L. Rohmawati, M. A. Nugroho, and Wagito, "Implementasi Chatbot Pada WhatsApp Untuk Monitoring Sumber Daya Server," Journal of Information System Management (JOISM), vol. 4, no. 2, pp. 107–112, 2023.

[2] F. H. P. Hari and F. Alfarisi, "Perancangan Chatbot WhatsApp untuk Mendukung Layanan Penerimaan Mahasiswa Baru Menggunakan Node.js," SISFOTENIKA, vol. 15, no. 2, pp. 101–114, 2025, doi: 10.30700/sisfotenika.v15i2.559.

[3] Z. Salsabila and A. Syarif, "Pemanfaatan Media Google Drive Dalam Pengelolaan Dokumen Elektronik Komisi Aparatur Sipil Negara," Jurnal Sekretari & Administrasi (Serasi), vol. 20, no. 2, pp. 116–125, 2023, doi: 10.36080/js.v20i2.2172.

[4] P. M. Pratisthaa, I. M. Sukarsaa, and P. W. Buana, "Implementasi Google Drive API untuk Upload, Sharing, dan Download Data pada Aplikasi Berbasis Web," JITTER – Jurnal Ilmiah Teknologi dan Komputer, 2022.

[5] A. G. Aidyn, A. A. Kurniawan, B. Prawiradijaya, and D. R. Aryaputra, "Integrasi Layanan Cloud Storage dengan WhatsApp Bot Menggunakan Webhook dan Firebase," Universitas Sebelas Maret, 2025.

- [6] R. Zalsahra, H. Sujadi, and Suhendri, "Perancangan dan Implementasi Bot WhatsApp Otomatis Menggunakan Node JS dan Baileys Berbasis Website (Studi Kasus: Toko Untitled)," Seminar Teknologi Majalengka (STIMA), 2025.
- [7] K. Patil, "Authentication and Authorization in Web Applications," *Journal of Engineering and Applied Sciences Technology (JEAST)*, vol. 5, no. 1, pp. 1–8, Feb. 2023.
- [8] D. Hardt, "The OAuth 2.0 Authorization Framework," RFC 6749, Internet Engineering Task Force (IETF), Oct. 2012. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc6749>
- [9] S. AlGhozali and S. Mukminatun, "Natural Language Processing of Gemini Artificial Intelligence Powered Chatbot," *Balankas: An International Multidisciplinary Research Journal*, vol. 1, no. 1, pp. 41–48, 2024, doi: 10.66317/688570.
- [10] M. Imran and N. Almusharraf, "Google Gemini as a Next-Generation AI Educational Tool: A Review of Emerging Educational Technology," *Smart Learning Environments*, vol. 11, no. 1, p. 22, 2024, doi: 10.1186/s40561-024-00310-z.
- [11] I. Amirulloh, M. W. Pertiwi, and T. Wibisono, "Rancang Bangun Chatbot WhatsApp Menggunakan Node JS dan Model Natural Language Processing untuk Layanan PPDB SMK YPC Tasikmalaya," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3846.
- [12] I. K. D. Senapatha, "Implementasi Single Sign-On Menggunakan Google Identity, REST dan OAuth 2.0 Berbasis Scrum," *JuTISI*, vol. 7, no. 2, pp. 307–320, Aug. 2021, doi: 10.28932/jutisi.v7i2.3437.
- [13] P. R. Aghnia, A. Rusmana, and A. S. Rohman, "Pembuatan Google Drive Sebagai Tempat Penyimpanan Arsip Digital Terpusat Di PT. Karya Dua Anyam," *Jurnal Ilmiah Multidisiplin*, vol. 2, no. 3, pp. 52–58, 2023, doi: 10.56127/jukim.v2i03.803.
- [14] E. Nurhayati and A. Agussalim, "Rancang Bangun Back-end API pada Aplikasi Mobile AyamHub Menggunakan Framework Node JS Express," *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, vol. 11, no. 3, pp. 524–531, Jul. 2023, doi: 10.26418/justin.v11i3.66823.
- [15] R. Kurniah, "Penerapan Sistem Informasi Berbasis Web Untuk Pengelolaan dan Pengarsipan Dokumen," *Infotek: Jurnal Informatika dan Teknologi*, vol. 6, no. 2, pp. 258–267, 2023, doi: 10.29408/jit.v6i2.11946.