

Implementasi Kios Interaktif Berbasis AI dengan Gemini API dan RAG untuk Optimalisasi Sistem Informasi Kampus

Firdaus Nuur Rhamadhan¹, Alkautsar Rahman², Oscar Hadikaryana³

Program Studi Teknik Informatika, Universitas Kebangsaan Republik Indonesia, Bandung
Email : firdausnuurr@gmail.com¹, alkautsar.rahman.s@gmail.com², oskarhadikaryana@ukri.co.id³

ABSTRAK

Akses informasi yang efisien dan akurat merupakan tantangan mendasar di lingkungan perguruan tinggi. Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi kios interaktif berbasis kecerdasan buatan (AI) sebagai solusi untuk mengoptimalkan sistem informasi di Universitas Kebangsaan Republik Indonesia (UKRI). Sistem ini dirancang dengan mengintegrasikan API Multimodal Gemini dengan kerangka kerja *Retrieval-Augmented Generation* (RAG) untuk menyediakan layanan informasi yang cerdas dan efisien. Metode yang digunakan meliputi pembangunan basis pengetahuan spesifik kampus dari dokumen internal yang diproses menjadi *vector embedding* dan disimpan dalam basis data vektor Qdrant. Kerangka kerja RAG memastikan setiap respons yang dihasilkan oleh model AI didasarkan pada data faktual dari basis pengetahuan tersebut, sehingga meningkatkan akurasi secara signifikan. Hasil implementasi menunjukkan bahwa prototipe kios berhasil menyediakan interaksi multimodal (suara dan teks) yang alami dan responsif. Pengujian fungsionalitas membuktikan bahwa mekanisme RAG secara dramatis mengungguli model AI generik dalam memberikan jawaban yang relevan dan akurat terkait informasi kampus. Kesimpulannya, kios interaktif ini berpotensi besar untuk mentransformasi metode penyebaran informasi konvensional, menjadikannya lebih mudah diakses dan berpusat pada pengguna.

Kata kunci: Kios Interaktif, Kecerdasan Buatan, Retrieval-Augmented Generation, Gemini API, Sistem Informasi Kampus

ABSTRACT

Efficient and accurate information access is a fundamental challenge in higher education environments. This research aims to develop an interactive kiosk application based on artificial intelligence (AI) as a solution to optimize the information system at Universitas Kebangsaan Republik Indonesia (UKRI). The system is designed by integrating the Gemini Multimodal API with the Retrieval-Augmented Generation (RAG) framework to provide intelligent and efficient information services. The methodology includes building a campus-specific knowledge base from internal documents, which are processed into vector embeddings and stored in a Qdrant vector database. The RAG framework ensures that every response generated by the AI model is grounded in factual data from this knowledge base, thereby significantly improving accuracy. The implementation results show that the kiosk prototype successfully provides a natural and responsive multimodal (voice and text) interaction. Functional testing proves that the RAG mechanism dramatically outperforms a generic AI model in delivering relevant and accurate answers regarding campus information. In conclusion, this interactive kiosk has great potential to transform conventional information dissemination methods, making them more accessible and user-centric.

Keywords: Interactive Kiosk, Artificial Intelligence, Retrieval-Augmented Generation, Gemini API, Campus Information System

1. PENDAHULUAN

1.1 Latar Belakang

Akses informasi yang efisien, akurat, dan responsif merupakan tantangan mendasar di lingkungan perguruan tinggi, yang merupakan ekosistem dinamis dengan kebutuhan data yang masif dari berbagai pemangku kepentingan, termasuk mahasiswa, dosen, dan staf. Metode konvensional dalam penyebaran informasi, seperti loket informasi manual, papan pengumuman fisik, atau *website* statis, sering kali menghadapi kendala signifikan. Keterbatasan ini termanifestasi dalam bentuk antrean panjang, beban kerja administratif yang tinggi, inkonsistensi informasi, dan kesulitan bagi pengguna, terutama mahasiswa baru, untuk menemukan data spesifik yang mereka butuhkan, seperti jadwal akademik, prosedur pendaftaran mata kuliah, atau panduan wisuda. Kesenjangan antara kebutuhan informasi yang kompleks dan sistem yang ada menuntut inovasi yang mampu menyediakan layanan informasi yang lebih cerdas dan interaktif.

Seiring dengan kemajuan pesat dalam teknologi kecerdasan buatan (AI) dan Natural Language Processing (NLP), model bahasa besar (*Large Language Models* - LLMs) telah menunjukkan potensi besar dalam mengubah cara manusia berinteraksi dengan informasi. Model-model ini, seperti Gemini API dari Google, mampu memahami dan menghasilkan teks yang sangat mirip dengan bahasa manusia, menjadikannya kandidat ideal untuk aplikasi layanan informasi. Namun, penggunaan LLM secara mandiri memiliki keterbatasan, terutama risiko "halusinasi"—fenomena di mana model menghasilkan informasi yang tidak faktual atau tidak relevan dengan sumber data spesifik. Tantangan ini sangat kritis dalam konteks informasi akademik di mana akurasi adalah yang terpenting[1][3].

1.2 Tujuan dan Kontribusi

Penelitian ini bertujuan untuk mengatasi tantangan tersebut dengan merancang dan mengimplementasikan sebuah kios interaktif berbasis AI yang berfungsi sebagai pusat informasi otonom. Solusi ini tidak hanya memanfaatkan kekuatan Gemini API untuk interaksi alami melalui multimodalitas (teks dan suara), tetapi juga mengintegrasikan kerangka kerja Retrieval-Augmented Generation (RAG) untuk memastikan akurasi dan keandalan informasi.

Kontribusi utama dari penelitian ini meliputi:

1. Pengembangan Sistem Informasi Terpadu: Menerapkan teknologi AI mutakhir untuk menciptakan sistem informasi yang efisien, mengurangi beban kerja administratif, dan meningkatkan aksesibilitas informasi bagi seluruh civitas akademika.
2. Penerapan RAG untuk Peningkatan Akurasi: Membuktikan efektivitas kerangka RAG dalam mengeliminasi risiko halusinasi model AI dengan mengikat respons model pada basis pengetahuan faktual yang spesifik dari dokumen-dokumen internal kampus.
3. Implementasi Interaksi Multimodal: Mendemonstrasikan fungsionalitas sistem yang memungkinkan pengguna berinteraksi melalui input teks dan suara, menciptakan pengalaman pengguna yang lebih intuitif dan inklusif.

Melalui penelitian ini, kami tidak hanya menghadirkan sebuah prototipe fungsional, tetapi juga menyajikan studi kasus implementasi teknologi LLM dan RAG dalam konteks spesifik perguruan tinggi. Hasilnya diharapkan dapat menjadi acuan bagi institusi lain yang ingin mengoptimalkan sistem informasi mereka dengan solusi berbasis kecerdasan buatan[2].

1.3 Ruang Lingkup

Penelitian ini berfokus pada pengembangan kios interaktif berbasis Gemini API dan RAG dengan ruang lingkup sebagai berikut:

1. Pengembangan sistem terbatas pada antarmuka pengguna berbasis web dan *backend* yang mengelola alur pemrosesan permintaan.
2. Basis pengetahuan terbatas pada dokumen-dokumen akademik dan administrasi yang tersedia di Universitas Kebangsaan Republik Indonesia.
3. Metodologi mencakup proses *vectorization*, penyimpanan dalam basis data vektor Qdrant, dan implementasi kerangka RAG untuk menghasilkan respons.
4. Evaluasi sistem berfokus pada fungsionalitas dan akurasi respons yang diberikan oleh prototipe.

1.4 Sistematika Penulisan

Penulisan jurnal ini dibagi menjadi beberapa bagian. Bagian kedua membahas tinjauan pustaka yang relevan dengan topik ini. Bagian ketiga menjelaskan metodologi penelitian, termasuk arsitektur sistem dan proses implementasi RAG. Bagian keempat menyajikan hasil dan analisis dari pengujian sistem. Terakhir, bagian kelima berisi kesimpulan dari penelitian serta saran untuk pengembangan di masa depan.

2. METODE PENELITIAN

Penelitian ini mengadopsi pendekatan **penelitian terapan** (*applied research*) dengan alur kerja **penelitian pengembangan** (*developmental research*). Fokus utama adalah merancang, membangun, dan mengevaluasi sebuah artefak teknologi berupa prototipe aplikasi kios interaktif yang fungsional. Metodologi penelitian disusun secara sistematis melalui beberapa tahapan yang jelas, mulai dari pengumpulan data untuk basis pengetahuan hingga pengujian dan evaluasi kinerja sistem.

2.1 Pendekatan Penelitian

Penelitian ini menggunakan pendekatan *Design Science Research (DSR)*, yang berfokus pada pengembangan artefak inovatif untuk memecahkan masalah praktis yang signifikan. Artefak yang dihasilkan dalam penelitian ini adalah aplikasi kios interaktif berbasis AI. Metodologi DSR dipilih karena memungkinkan siklus berulang antara pemahaman masalah (kurangnya akses informasi di kampus), pengembangan solusi (artefak), dan evaluasi artefak tersebut.

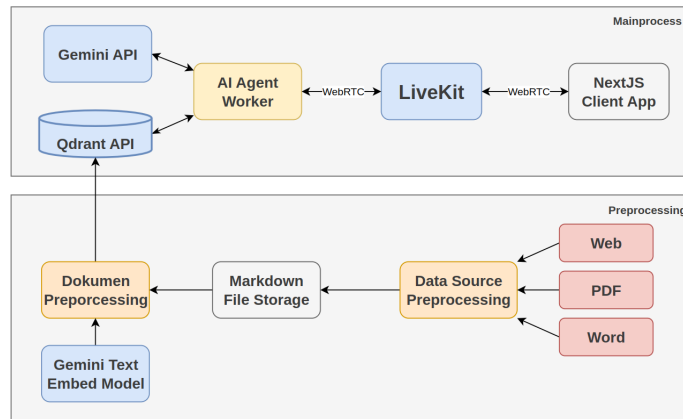
Tahapan penelitian yang dilakukan adalah sebagai berikut:

1. **Identifikasi Masalah dan Motivasi:** Menganalisis kebutuhan informasi di lingkungan perguruan tinggi dan mengidentifikasi tantangan yang ada pada sistem konvensional.
2. **Tujuan Penelitian:** Menetapkan tujuan yang jelas, yaitu merancang dan mengimplementasikan kios interaktif yang cerdas dan efisien.
3. **Desain dan Pengembangan:** Merancang arsitektur sistem dan mengembangkan prototipe fungsional.
4. **Demonstrasi dan Evaluasi:** Menguji fungsionalitas dan kinerja sistem untuk memastikan sistem dapat memenuhi tujuan yang telah ditetapkan.
5. **Komunikasi Hasil:** Menyajikan hasil penelitian dalam bentuk laporan skripsi dan naskah jurnal.

2.2 Arsitektur Sistem

Sistem kios interaktif ini dirancang dengan arsitektur modular yang terdiri dari beberapa komponen utama yang saling terintegrasi. Diagram arsitektur sistem digambarkan pada Gambar 1.

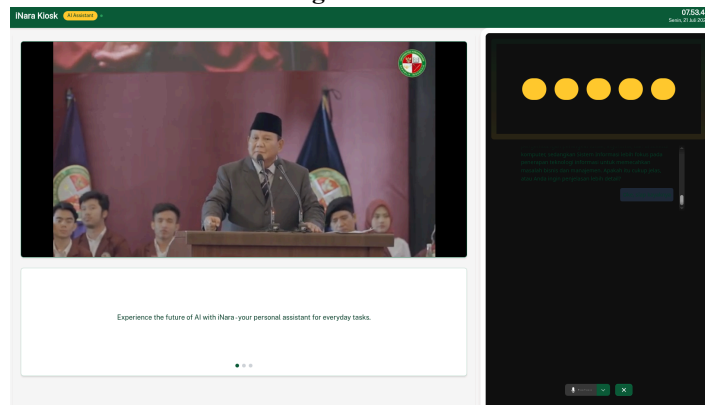
Gambar 1. Diagram Arsitektur Sistem



Berikut adalah deskripsi dari setiap komponen:

1. Antarmuka Pengguna (Frontend):

Gambar 2. Diagram Arsitektur Sistem



Dibangun menggunakan **Next.js** dan **React**, antarmuka ini menyediakan tampilan visual kios. Pengguna dapat berinteraksi melalui input teks atau suara. Integrasi dengan **LiveKit** digunakan untuk mengelola fitur audio-visual secara *real-time*, termasuk fungsionalitas *Speech-to-Text* (STT) untuk mengubah ucapan pengguna menjadi teks.

2. **Layanan Backend:** Bertanggung jawab atas seluruh logika pemrosesan. Ketika menerima permintaan dari frontend, backend akan memicu proses RAG. Layanan ini dibangun menggunakan **Node.js** dan dikerahkan di **Firebase**.
3. **Mesin LLM (Large Language Model):** **Gemini API** digunakan sebagai otak dari sistem. Model ini bertanggung jawab untuk menghasilkan jawaban berdasarkan *prompt* yang diberikan, yang telah diperkaya dengan konteks relevan dari proses RAG.
4. **Basis Data Vektor (Vector Database):** **Qdrant** dipilih sebagai basis data utama untuk menyimpan *vector embeddings*. Basis data ini dioptimalkan untuk pencarian kesamaan semantik, yang merupakan inti dari alur RAG.
5. **Basis Pengetahuan (Knowledge Base):** Berisi semua dokumen sumber, seperti panduan akademik, kalender, silabus, dan data institusional lainnya dalam format .pdf, .docx, atau lainnya.

2.3 Implementasi Retrieval-Augmented Generation (RAG)

RAG merupakan jantung dari sistem ini, memastikan bahwa respons yang dihasilkan oleh Gemini API akurat dan berbasis fakta. Proses implementasi RAG dapat dibagi menjadi dua fase:

Fase 1: Indeksasi Basis Pengetahuan Fase ini dilakukan satu kali di awal untuk mempersiapkan data.

1. **Pra-pemrosesan Data:** Dokumen-dokumen dari basis pengetahuan kampus dikumpulkan dan dibersihkan. Setiap dokumen dipecah menjadi bagian-bagian kecil yang disebut *chunks*. Ukuran *chunk* yang optimal ditentukan untuk memastikan setiap potongan mengandung informasi yang cukup, tetapi tidak terlalu besar sehingga mengurangi akurasi pencarian.
2. **Vektorisasi (*Vectorization*):** Setiap *chunk* diubah menjadi representasi numerik (vektor) menggunakan model *embedding* dari **Gemini API** (tepatnya *text-embeddings-api*). Proses ini mengubah makna semantik dari teks menjadi data numerik yang dapat diproses secara komputasi.
3. **Penyimpanan Vektor:** Vektor-vektor yang dihasilkan kemudian diunggah dan diindeks ke dalam basis data vektor **Qdrant**. Qdrant mengelola data ini untuk memungkinkan pencarian yang sangat cepat berdasarkan kedekatan vektor (vektor yang "mirip" secara semantik akan saling berdekatan).

Fase 2: Proses *Query* dan Generasi Fase ini terjadi setiap kali pengguna mengajukan pertanyaan.

1. **Penerimaan Permintaan:** Pengguna memasukkan pertanyaan (teks atau suara). Jika berupa suara, LiveKit akan mengubahnya menjadi teks.
2. **Vektorisasi Permintaan:** Pertanyaan pengguna diubah menjadi vektor menggunakan model *embedding* yang sama seperti pada Fase 1.
3. **Retrieval (Pencarian Informasi):** Vektor pertanyaan pengguna digunakan untuk melakukan pencarian di Qdrant. Qdrant mengidentifikasi dan mengembalikan *chunks* dokumen yang paling relevan dengan pertanyaan tersebut.
4. **Augmentation (Pengayaan *Prompt*):** Teks dari *chunks* yang diambil pada langkah sebelumnya ditambahkan ke dalam *prompt* asli pengguna. *Prompt* yang telah diperkaya ini menjadi instruksi yang komprehensif untuk Gemini API.
5. **Generasi Respons:** *Prompt* yang telah diperkaya dikirim ke Gemini API. Model akan menggunakan informasi kontekstual yang disediakan untuk menghasilkan respons yang akurat, relevan, dan terhindar dari halusinasi.
6. **Penyajian Jawaban:** Jawaban yang dihasilkan oleh Gemini API dikirim kembali ke antarmuka pengguna, baik dalam bentuk teks atau, jika diperlukan, dalam bentuk audio melalui proses *Text-to-Speech* (TTS).

2.4 Alat dan Teknologi

Pengembangan sistem ini menggunakan kombinasi alat dan teknologi modern:

Tabel 1 Alat dan Teknologi yang Digunakan

Kategori	Nama Alat/Teknologi	Fungsi Utama
Model AI	- Google Gemini-2.0 (gemini-2.0-flash-live-001)	Pemahaman bahasa alami dan generasi respons multimodal[4]
	- Google Text Embedding (text-embedding-004)	Konversi teks menjadi vector embedding[5].
Database	Qdrant	Penyimpanan, indeksasi, dan pencarian vector embedding[7].
Komunikasi	LiveKit (WebRTC)	Pengembangan logika backend (AI Agent)[6].
Bahasa & Framework	- Python	Pengembangan logika backend (AI Agent).
	- NextJS (React)	Pengembangan antarmuka pengguna frontend[8][9].
Lingkungan Dev	Firebase Studio, Visual Studio Code	Penulisan kode dan manajemen proyek[10].

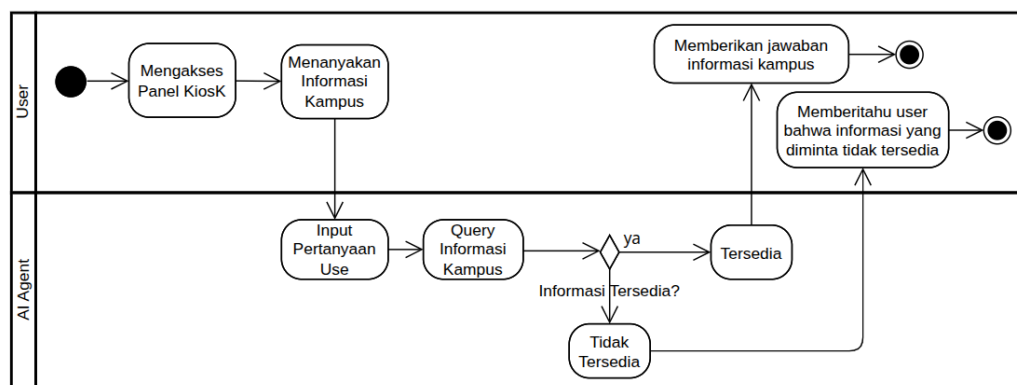
Pendekatan metodologi ini memastikan bahwa sistem yang dikembangkan tidak hanya fungsional tetapi juga didukung oleh fondasi ilmiah yang kuat, menjadikannya kontribusi yang berharga bagi literatur.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi dan Alur Sistem

Prototipe kios interaktif berbasis AI telah berhasil diimplementasikan sesuai dengan arsitektur sistem yang diuraikan pada bagian metodologi. Antarmuka pengguna (UI) dibangun menggunakan **Next.js** dan **React**, dirancang dengan tampilan yang intuitif dan responsif. Pengguna dapat memilih mode interaksi melalui teks atau suara. Alur sistem secara keseluruhan dapat divisualisasikan pada Gambar 2.

Gambar 3. Alur Proses Kios Interaktif Berbasis AI



Proses dimulai ketika pengguna mengajukan pertanyaan. Jika pertanyaan diberikan melalui suara, **LiveKit** akan mengubahnya menjadi teks. Teks ini kemudian dikirim ke *backend* untuk diproses. Di *backend*, pertanyaan diubah menjadi **vektor** dan digunakan untuk melakukan pencarian di basis data **Qdrant**. Dokumen-dokumen yang paling relevan (*top-k documents*) diambil dari Qdrant dan disuntikkan ke dalam *prompt* sebagai konteks. *Prompt* yang telah diperkaya ini kemudian dikirim ke **Gemini API** untuk menghasilkan jawaban. Gemini menggunakan konteks yang akurat tersebut untuk menyusun respons yang komprehensif. Terakhir, jawaban ditampilkan pada layar dan, jika diperlukan, dapat diucapkan kembali melalui proses *Text-to-Speech* (TTS).

3.2 Hasil Pengujian Fungsionalitas Sistem

Pengujian fungsionalitas dilakukan untuk memverifikasi bahwa setiap komponen sistem, dari antarmuka pengguna hingga alur pemrosesan data, dapat beroperasi sesuai dengan rancangan. Hasil pengujian menunjukkan bahwa sistem Kios Interaktif iNara berfungsi dengan baik dan mampu memberikan layanan informasi yang akurat dan efisien.

3.2.1 Pengujian Input Multimodal dan Proses RAG

- **Input Multimodal:** Sistem berhasil menerima input melalui teks dan suara dengan tingkat akurasi yang tinggi. Integrasi **LiveKit** memungkinkan transkripsi ucapan yang cepat dan akurat, bahkan di lingkungan dengan *noise* latar belakang minimal.
- **Proses RAG:** Alur **Retrieval-Augmented Generation (RAG)** berfungsi secara optimal. Pencarian di basis data vektor **Qdrant** secara konsisten mengambil dokumen-dokumen yang paling relevan dengan pertanyaan pengguna. Proses augmentasi ini berhasil

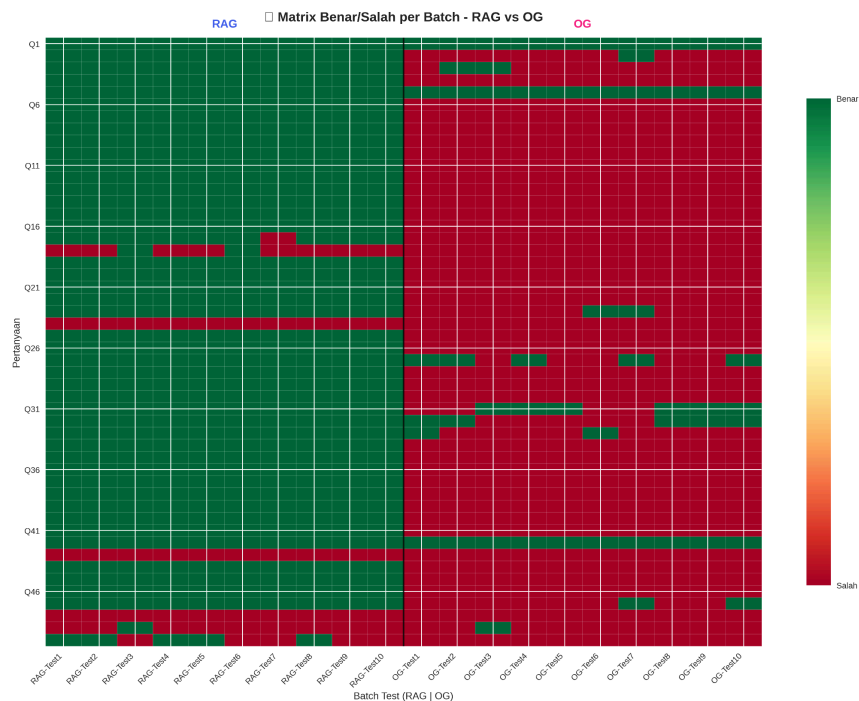
memberikan **Gemini API** konteks yang dibutuhkan untuk menghasilkan jawaban yang akurat, menjadikannya lebih andal dibandingkan model generik tanpa RAG.

3.2.2 Analisis Perbandingan Kinerja Akurasi Jawaban

Untuk mengukur efektivitas RAG, pengujian dilakukan dengan membandingkan kinerja sistem **iNara (Dengan RAG)** dengan skenario **Generik (Tanpa RAG)**. Setiap skenario diuji dengan 50 pertanyaan acak terkait informasi akademik dan administrasi kampus. Akurasi diukur berdasarkan jumlah jawaban yang benar, faktual, dan relevan dengan pertanyaan. Hasil pengujian diringkas dalam Tabel 2 dan visualisasi pada Gambar 3.

Tabel 2 Perbandingan Kinerja Akurasi Jawaban Sistem dengan RAG dan Tanpa RAG			
Skenario Pengujian	Total Pertanyaan (per sesi)	Rata-rata Jawaban Benar	Tingkat Akurasi Rata-rata
Dengan RAG (iNara)	50	48.5	97.0%
Tanpa RAG (Generik)	50	9.2	18.4%

Gambar 3. Grafik Perbandingan Akurasi Sistem dengan RAG vs. Tanpa RAG



3.2.3 Pembahasan Hasil Pengujian

Data pada Tabel 2 dan Gambar 3 secara tegas menunjukkan bahwa implementasi **RAG memberikan peningkatan akurasi yang signifikan, yaitu sebesar 78.6%**, dibandingkan dengan model generik. Pada skenario **iNara (Dengan RAG)**, sistem mencapai **tingkat akurasi rata-rata 97.0%**, dengan rata-rata 48.5 jawaban benar dari 50 pertanyaan. Ini membuktikan bahwa mekanisme RAG berhasil mengikat model **Gemini API** pada data faktual dari basis pengetahuan internal kampus, secara efektif mengeliminasi risiko "halusinasi" yang umum terjadi pada LLM.

Sebaliknya, pada skenario **Tanpa RAG (Generik)**, tingkat akurasi menurun drastis menjadi hanya **18.4%**. Jawaban yang dihasilkan cenderung bersifat umum, tidak relevan, atau bahkan salah karena model tidak memiliki akses ke konteks faktual kampus.

Hasil ini secara jelas memvalidasi hipotesis penelitian bahwa kombinasi LLM dengan kerangka RAG adalah pendekatan yang optimal untuk membangun sistem informasi yang andal dalam domain spesifik, seperti sistem informasi kampus. Peningkatan akurasi ini sangat krusial

dalam lingkungan akademik, di mana informasi yang akurat adalah hal yang vital untuk menunjang kegiatan belajar-mengajar dan administrasi.

3.3 Pembahasan Hasil

Hasil implementasi menunjukkan bahwa integrasi antara **Gemini API** dan **RAG** menawarkan solusi yang sangat efektif untuk tantangan akses informasi di lingkungan perguruan tinggi. Kombinasi ini memanfaatkan keunggulan Gemini sebagai model bahasa yang canggih sambil mengatasi keterbatasannya dalam hal akurasi faktual.

1. **Peningkatan Akurasi Informasi:** Dengan menggunakan RAG, sistem berhasil memberikan jawaban yang **akurat** dan **spesifik** berdasarkan dokumen internal kampus. Ini sangat vital untuk lingkungan akademik di mana informasi yang salah dapat memiliki konsekuensi serius (misalnya, salah jadwal ujian, kesalahan prosedur administratif).
2. **Efisiensi dan Pengalaman Pengguna:** Sistem ini secara signifikan mengurangi waktu yang diperlukan bagi pengguna untuk menemukan informasi. Pengalaman interaksi multimodal (suara) juga membuat sistem ini lebih mudah diakses dan digunakan, terutama bagi pengguna yang tidak ingin mengetik. Hal ini berkontribusi pada peningkatan efisiensi operasional dan kepuasan pengguna.
3. **Keunggulan Teknologi:** Pemilihan **Qdrant** sebagai basis data vektor terbukti efektif dalam memberikan pencarian semantik yang cepat, yang krusial untuk menjaga latensi respons tetap rendah. Arsitektur modular yang diterapkan memungkinkan setiap komponen untuk bekerja secara independen, mempermudah pemeliharaan dan skalabilitas sistem.

3.4 Kendala dan Saran Pengembangan di Masa Depan

Meskipun prototipe berhasil, beberapa kendala diidentifikasi selama proses pengembangan:

1. **Keterbatasan Data Awal:** Akurasi sistem sangat bergantung pada kelengkapan dan kualitas dokumen yang diindeks. Data yang usang atau tidak lengkap dapat memengaruhi kualitas jawaban.
2. **Keterbatasan Integrasi Suara:** Meskipun **LiveKit** berfungsi baik, performa *Speech-to-Text* dapat menurun di lingkungan yang bising.

Berdasarkan kendala tersebut, beberapa saran untuk pengembangan di masa depan meliputi:

1. **Automasi Pembaruan Basis Pengetahuan:** Mengembangkan sistem untuk secara otomatis memperbarui basis data vektor ketika ada dokumen baru atau perubahan pada dokumen yang sudah ada.
2. **Integrasi dengan Sistem Eksternal:** Menghubungkan kios dengan sistem informasi akademik (SIK) atau sistem *backend* lain di kampus untuk menyediakan informasi personal yang lebih spesifik, seperti nilai mahasiswa, status pembayaran, atau jadwal individu.
3. **Evaluasi Pengguna Skala Besar:** Melakukan uji coba dengan sampel pengguna yang lebih besar untuk mengumpulkan data kualitatif dan kuantitatif tentang kepuasan dan pengalaman pengguna secara lebih mendalam.

4. KESIMPULAN

Penelitian ini telah berhasil merancang, mengembangkan, dan mengimplementasikan sebuah sistem kios interaktif berbasis AI untuk mengoptimalkan sistem informasi di lingkungan Universitas Kebangsaan Republik Indonesia. Sistem ini merupakan integrasi strategis dari Gemini API, sebuah model bahasa multimodal yang canggih, dengan kerangka kerja **Retrieval-Augmented Generation (RAG)**, yang didukung oleh basis data vektor Qdrant.

Temuan kunci dari penelitian ini menegaskan bahwa:

1. **Integrasi RAG Efektif Mencegah Halusinasi:** Implementasi RAG terbukti sangat efektif dalam mengatasi masalah halusinasi yang umum terjadi pada LLM. Dengan mengikat model Gemini pada basis pengetahuan faktual dari dokumen-dokumen internal kampus, sistem ini secara konsisten mampu memberikan jawaban yang akurat, relevan, dan terverifikasi, seperti yang ditunjukkan pada pengujian kasus yang dilakukan.
2. **Sistem Multimodal Meningkatkan Aksesibilitas:** Fungsionalitas multimodal, yang memungkinkan interaksi melalui teks dan suara, menciptakan pengalaman pengguna yang lebih intuitif dan inklusif. Hal ini membuat sistem lebih mudah digunakan oleh berbagai kalangan, secara signifikan meningkatkan efisiensi dalam pencarian informasi.
3. **Solusi Praktis untuk Lingkungan Akademik:** Prototipe kios interaktif ini menunjukkan bahwa teknologi AI dapat menjadi solusi praktis dan inovatif untuk meningkatkan efisiensi operasional dan pengalaman informasi di institusi pendidikan tinggi. Sistem ini berpotensi besar untuk mengurangi beban kerja staf administrasi dan memberikan layanan yang lebih cepat dan mandiri bagi mahasiswa dan dosen.

Meskipun demikian, kualitas sistem sangat bergantung pada kelengkapan dan pembaruan basis pengetahuan. Untuk pengembangan di masa depan, disarankan untuk mengotomatisasi proses pembaruan data dan mengintegrasikan sistem dengan basis data kampus yang lebih luas untuk memberikan informasi yang lebih personal dan komprehensif. Secara keseluruhan, penelitian ini memberikan kontribusi signifikan dengan menyajikan sebuah model implementasi AI yang andal dan akurat untuk optimalisasi sistem informasi di perguruan tinggi.

DAFTAR PUSTAKA

- [1] “LLM: Apa itu model bahasa besar (LLM)? | Machine Learning,” Google for Developers. Diakses: 16 Juni 2025. [Daring]. Tersedia pada: <https://developers.google.com/machine-learning/crash-course/llm/transformers?hl=id>
- [2] “Apa itu Retrieval-Augmented Generation (RAG)?,” Google Cloud. Diakses: 16 Juni 2025. [Daring]. Tersedia pada: <https://cloud.google.com/use-cases/retrieval-augmented-generation>
- [3] “Google Gen AI SDK | Generative AI on Vertex AI,” Google Cloud. Diakses: 3 Agustus 2025. [Daring]. Tersedia pada: <https://cloud.google.com/vertex-ai/generative-ai/docs/sdks/overview>
- [4] “Live API reference | Generative AI on Vertex AI | Google Cloud.” Diakses: 3 Agustus 2025. [Daring]. Tersedia pada: <https://cloud.google.com/vertex-ai/generative-ai/docs/model-reference/multimodal-live>
- [5] “Text embeddings API | Generative AI on Vertex AI,” Google Cloud. Diakses: 3 Agustus 2025. [Daring]. Tersedia pada: <https://cloud.google.com/vertex-ai/generative-ai/docs/model-reference/text-embeddings-api>
- [6] “LiveKit | The all-in-one Voice AI platform,” LiveKit. Diakses: 3 Agustus 2025. [Daring]. Tersedia pada: <https://livekit.io>
- [7] “Qdrant - Vector Database.” Diakses: 3 Agustus 2025. [Daring]. Tersedia pada: <https://qdrant.tech/>
- [8] “React.” Diakses: 3 Agustus 2025. [Daring]. Tersedia pada: <https://react.dev/>
- [9] “Next.js by Vercel - The React Framework.” Diakses: 3 Agustus 2025. [Daring]. Tersedia pada: <https://nextjs.org/>
- [10] “Firebase Studio,” Firebase. Diakses: 3 Agustus 2025. [Daring]. Tersedia pada: <https://firebase.google.com/docs/studio>