Jurnal Informatika Terpadu

https://journal.nurulfikri.ac.id/index.php/JIT
ISSN ONLINE : 2460-8998

PERANCANGAN BALI *VIRTUAL TOURISM ASSISTANT* BERBASIS *ARTIFICIAL INTELLIGENCE* MENGGUNAKAN *NATURAL LANGUAGE PROCESSING*

I Putu Andika Subagya Putra[1], Wayan Andrika Putera[2], I Komang Gde Trisna Puwantara[3], Ni Komang Anggita Trisanti[4], I Gusti Agung Ngurah Eka Adnyana[5]

[1,4,5]Sistem Informasi, Universitas Markandeya
[2]Manajemen Informatika, Institut Teknologi dan Bisnis STIKOM Bali
[3]Manajemen Ekonomi, Universitas Markandeya
Bangli, Bali, Indonesia 80614
subagyaputra@gmail.com, andrika.putera@stikom-bali.ac.id, trisnapurwantara@markandeyabali.ac.id, anggibangli123@gmail.com, gungeka050@gmail.com

Abstract

Bali tourists require integrated information on attractions, operating hours, ticket tariffs, and temple etiquette, yet existing platforms remain fragmented while small local language models frequently hallucinate numerical and geographical facts in Indonesian. This study designs, implements, and evaluates the Bali Virtual Tourism Assistant, a web conversational system combining Natural Language Processing (NLP), Retrieval-Augmented Generation (RAG), and rule-based fact extraction. Developed via Waterfall with iterative testing, the system integrates Laravel, 318 Markdown documents across nine Bali regencies, 768-dimensional nomic-embed-text vectors, a ticket price detector (DestinationFacts), and local llama3.2:3b via Ollama. Black-box testing passed all 15 functional and three negative cases (100%). On 954 named queries from document titles, longest-name matching raised Hit@1 from 78.93% (pure vector) to 100.00%. On 100 varied free-text queries (partial names, nameless descriptions, typos, mixed Indonesian–English, and regional criteria), semantic vector search achieved 85.00% Hit@5 (MRR@5 = 0.7298), outperforming keyword search (73.00% Hit@5) by twofold on nameless descriptions (70.00% vs. 35.00%), while combined ranking reached 96.00% Hit@5. Paired evaluation (N = 20) showed RAG improved answer accuracy from 10.00% (without RAG) to 90.00% (18/20), completeness from 23.60% to 92.00%, and ticket accuracy from 8.33% (1/12) to 100.00% (12/12) with 6.50 s median response time.

Keywords: *Bali Tourism, Large Language Model, Natural Language Processing, Retrieval-Augmented Generation, Virtual Assistant*

Abstrak

Wisatawan Bali membutuhkan informasi terpadu mengenai destinasi, jam operasional, harga tiket, dan etika pura, namun platform yang ada masih terpisah sementara model bahasa berbobot ringan (*lightweight language model*) rentan mengalami halusinasi angka dan lokasi berbahasa Indonesia. Penelitian ini merancang, mengimplementasikan, dan menguji Bali *Virtual Tourism Assistant*, aplikasi percakapan web yang menggabungkan *Natural Language Processing* (NLP), *Retrieval-Augmented Generation* (RAG), dan pembacaan fakta harga tiket. Dikembangkan menggunakan metode *Waterfall* dengan perbaikan bertahap, sistem menghubungkan Laravel, 318 dokumen *Markdown* di sembilan kabupaten/kota Bali, vektor 768 dimensi *nomic-embed-text*, pendeteksi pertanyaan biaya (DestinationFacts), dan model bahasa llama3.2:3b melalui Ollama. Pengujian *black-box* meluluskan seluruh 15 skenario fungsional dan tiga skenario negatif (100%). Pada 954 pertanyaan bernama sesuai judul dokumen, prioritas nama terpanjang meningkatkan *Hit@1* dari 78,93% menjadi 100,00%. Pada 100 pertanyaan variasi bahasa (singkatan nama, deskripsi tanpa nama, salah ketik, bahasa campuran, kriteria wilayah), pencarian vektor mencapai *Hit@5* 85,00% (*MRR@5* = 0,7298), mengungguli kata kunci (*Hit@5* 73,00%) dua kali lipat pada deskripsi tanpa nama (70,00% vs. 35,00%), serta mencapai *Hit@5* 96,00% saat peringkat digabungkan. Evaluasi 20 pertanyaan membuktikan RAG meningkatkan ketepatan fakta dari 10,00% menjadi 90,00% (18/20), kelengkapan dari 23,60% ke 92,00%, dan ketepatan tiket dari 8,33% (1/12) ke 100,00% (12/12) dengan median respons 6,50 detik.

Kata kunci: Asisten Virtual, *Large Language Model* (LLM), Pariwisata Bali, Pemrosesan Bahasa Alami, *Retrieval-Augmented Generation*

Diterima 18 Oktober 2026, Direvisi 1 November 2026, Diterima untuk publikasi 24 November 2026

1. PENDAHULUAN

Pemulihan sektor pariwisata Bali pasca-pandemi mendorong lonjakan mobilitas kunjungan, dengan kunjungan wisatawan mancanegara langsung ke Bali naik dari 5,27 juta pada 2023 menjadi 6,33 juta pada 2024 dan sekitar 6,95 juta pada 2025, menurut BPS Provinsi Bali. Dalam merencanakan perjalanan, wisatawan membutuhkan informasi yang akurat pada setiap tahap kunjungan, mulai dari pemilihan daya tarik, pengecekan jam operasional dan tarif tiket masuk, hingga pemahaman aturan adat di kawasan suci pura [1], [2], [3]. Meskipun layanan peta digital dan ulasan kerumunan seperti Google Maps serta TripAdvisor unggul dalam navigasi geospasial dan peringkat populer, informasi mengenai aturan etika lokal, rincian tarif domestik/mancanegara, dan rekomendasi lintas kriteria masih tersebar di berbagai situs terpisah [4], [5]. Di sisi lain, penggunaan *Large Language Model* (LLM) komersial secara langsung tanpa penambatan sumber lokal berisiko menghasilkan halusinasi tarif, mencampuradukkan lokasi antar-kabupaten, atau memberikan panduan budaya yang keliru.

Teknologi *chatbot* berbasis *Natural Language Processing* (NLP), yakni cabang kecerdasan buatan (*Artificial Intelligence*/AI) yang membuat sistem komputasi dapat memproses, merepresentasikan makna semantik, dan membangkitkan bahasa alami manusia, menawarkan interaksi tanya-jawab terpadu dalam satu antarmuka percakapan [6], [7]. Perkembangan arsitektur agen percakapan telah bergeser dari pencocokan aturan statis menuju pemodelan representasi teks dan pembangkitan generatif yang adaptif terhadap variasi kalimat pengguna [8], [9], [10]. Pada konteks layanan informasi institusional di Indonesia, Suherlan dkk. [11] menunjukkan bahwa asisten percakapan mampu mempercepat akses informasi layanan publik, namun pendekatan berbasis aturan kaku masih terbatas ketika menghadapi variasi struktur kalimat bebas.

Pada domain pariwisata, sejumlah penelitian mutakhir telah mengeksplorasi integrasi agen percakapan dengan basis data wilayah. Handayani dkk. [12] membandingkan arsitektur retrieval dan generatif pada layanan informasi wisata, tetapi belum mengintegrasikan mekanisme penambatan dokumen sumber untuk mencegah fabrikasi angka biaya pada model generatif. Cruz dkk. [13] mengembangkan *chatbot* wisata Lisa untuk Kota Lisbon, Sperlí [14] merancang kerangka kerja warisan budaya berbasis pembelajaran mendalam, dan Benaddi dkk. [15] mengusulkan pabrik perangkat lunak untuk *chatbot* wisata cerdas, sementara tinjauan sistematis Prasanna dkk. [16] serta Tuyen dkk. [17] menyoroti perlunya pengendalian akurasi faktual pada sistem percakapan wisata. Celah penelitian (*research gap*) yang belum diselesaikan oleh studi-studi tersebut adalah ketiadaan arsitektur hibrida yang memadukan pencocokan entitas spesifik, pencarian vektor semantik, dan ekstraksi fakta biaya deterministik (*source-facts*) guna mengeliminasi halusinasi numerik pada model bahasa lokal berparameter kecil (3 miliar parameter) untuk korpus wisata berbahasa Indonesia.

Pendekatan *Retrieval-Augmented Generation* (RAG) menjembatani celah tersebut dengan mengambil dokumen relevan dari basis pengetahuan eksternal sebelum model bahasa menyusun jawaban [18], [19]. Tanpa penambatan konteks yang presisi, model generatif rentan memproduksi halusinasi intrinsik maupun ekstrinsik yang merusak keandalan informasi faktual [20], [21], [22]. Lebih jauh, studi *benchmark* RAG oleh Chen dkk. [23] mengungkapkan bahwa LLM berukuran kecil sangat peka terhadap derau konteks multi-dokumen dan sering gagal menyalin angka tarif secara konsisten apabila hanya mengandalkan generasi bebas. Oleh sebab itu, perancangan asisten wisata berbasis model bahasa berbobot ringan menuntut pemisahan tegas antara jalur pembangkitan naratif dan jalur ekstraksi atribut biaya terstruktur.

Keberhasilan RAG sangat bergantung pada ketepatan modul pencarian konteks dalam menjembatani kesenjangan leksikal dan semantik antara pertanyaan pengguna dan dokumen sumber [24]-[26]. Penelitian ini memanfaatkan model *nomic-embed-text* berdimensi 768 [27] yang dikombinasikan dengan aturan pencocokan nama dan kata kunci berbobot sebagai pencarian hibrida [28]. Tantangan krusial yang harus diatasi adalah isu bahasa: dokumentasi resmi Llama 3.2 hanya mencantumkan delapan bahasa utama tanpa menyertakan bahasa Indonesia secara eksplisit, sedangkan *nomic-embed-text* dilatih secara dominan pada korpus bahasa Inggris. Sebagaimana ditunjukkan oleh *benchmark* IndoNLU [29] dan IndoBERT [30], pemrosesan teks bahasa Indonesia di lapangan melibatkan variasi morfologi, singkatan tidak baku, serta alih kode bahasa Indonesia–Inggris (*code-mixing*) pada nama destinasi (misalnya "Rice Terrace" atau "Monkey Forest"), sehingga kemampuan model lintas-bahasa perlu dikompensasi dengan penambatan leksikal dan diuji secara empiris pada kueri berstrata.

Pemilihan arsitektur inferensi mandiri (*self-hosted deployment*) menggunakan model bahasa berbobot ringan llama3.2:3b (3,2 miliar parameter) melalui layanan Ollama didasarkan pada tiga pertimbangan rekayasa sistem [31], [32]. Pertama, fleksibilitas *deployment* mandiri memungkinkan sistem dioperasikan secara adaptif, baik pada peladen komputasi awan (*cloud server* / VPS) maupun server lokal (*on-premise*), sehingga memberikan kemandirian biaya operasional (*zero API cost*) bagi pengelola pariwisata daerah tanpa beban tarif per token ke penyedia API komersial pihak ketiga. Kedua, kendali peladen secara mandiri menjamin privasi dan kedaulatan data riwayat percakapan pengunjung di lingkungan server privat. Ketiga, efisiensi kuantisasi model (2,02 GB) memungkinkan inferensi berjalan optimal pada alokasi memori RAM 16 GB tanpa ketergantungan pada klaster multi-GPU komputasi awan berbiaya tinggi.

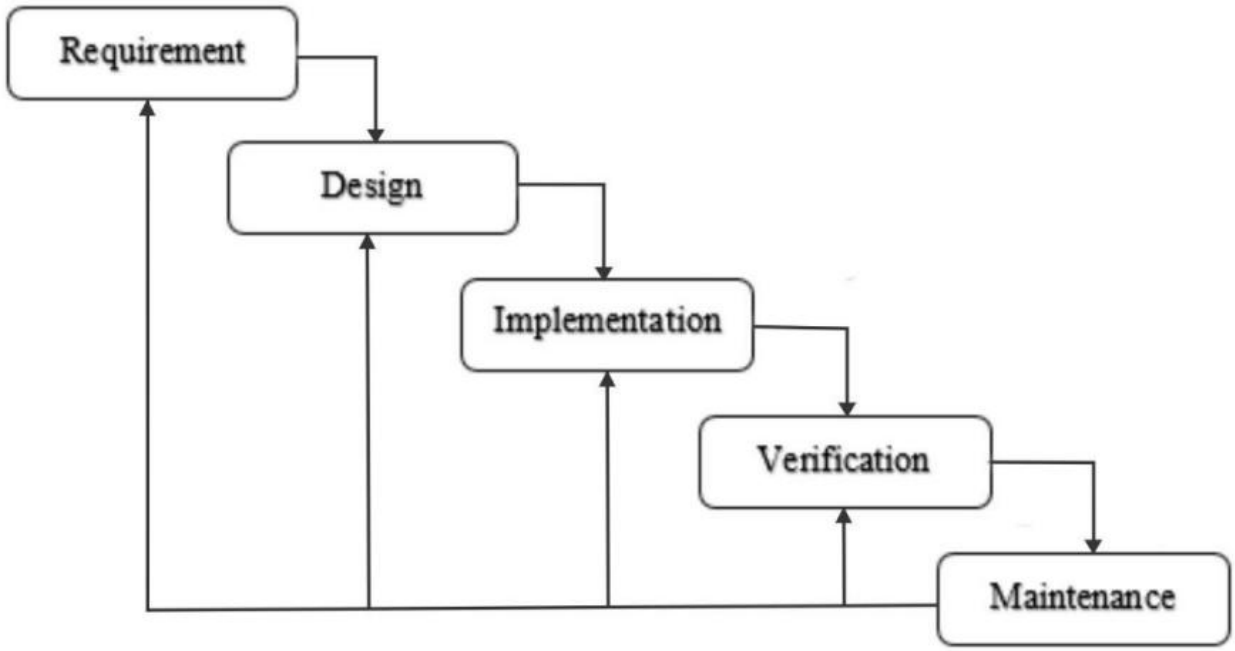
Berdasarkan latar belakang dan celah penelitian tersebut, penelitian ini berfokus pada tiga sasaran utama yang diuji secara terukur. Pertama, merancang dan mengimplementasikan arsitektur Bali *Virtual Tourism Assistant* berbasis NLP-RAG hibrida guna memenuhi seluruh spesifikasi kebutuhan fungsional dan nonfungsional dengan target kelulusan uji *black-box* 100%. Kedua, mengukur kontribusi masing-masing komponen pencarian, yang meliputi aturan pencocokan nama, kata kunci berbobot, vektor semantik *nomic-embed-text*, dan fusi peringkat, terhadap akurasi penemuan konteks pada kueri bernama maupun 100 kueri independen berstrata dengan target capaian *Hit@5* minimal 80%. Ketiga, membuktikan efektivitas integrasi RAG dan pengarah deterministik *source-facts* dalam meningkatkan kesetiaan fakta (*faithfulness*) hingga minimal 85% serta menjaga konsistensi tarif tiket 100% dibandingkan model dasar llama3.2:3b tanpa RAG.

2. METODE PENELITIAN

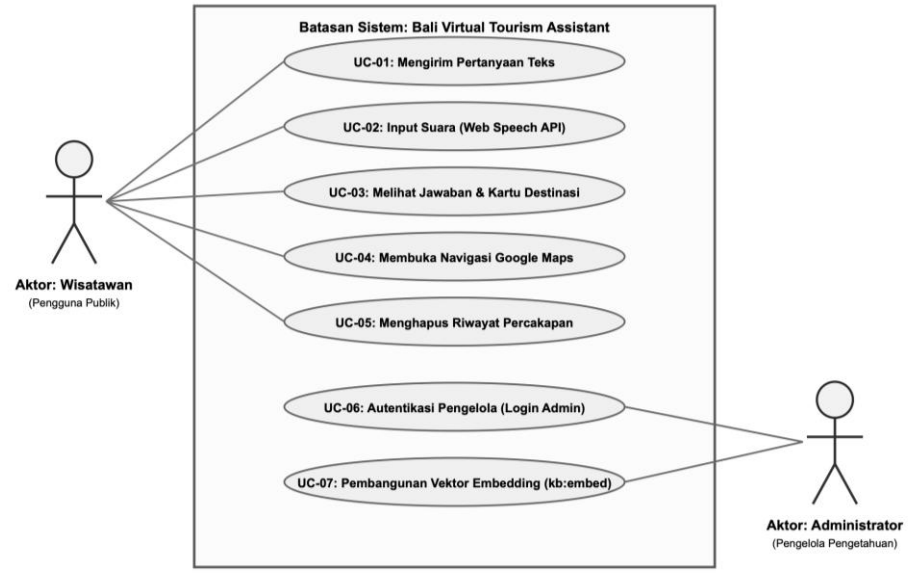
2.1 Tahapan Pengembangan dan Analisis Kebutuhan

Pengembangan perangkat lunak mengadopsi model siklus hidup *Waterfall* menurut Sommerville [33] yang mencakup analisis dan definisi kebutuhan, perancangan sistem dan perangkat lunak, implementasi dan pengujian unit, integrasi dan pengujian sistem, serta operasi dan pemeliharaan terbatas. Mengingat sistem yang dibangun mengintegrasikan komponen model bahasa probabilistik yang memerlukan penyetelan parameter dan penanganan halusinasi secara empiris, tahapan pengujian dan implementasi dilengkapi dengan siklus umpan balik verifikasi-koreksi sebagaimana direkomendasikan oleh Amershi dkk. [34] dalam rekayasa perangkat lunak berbasis pembelajaran mesin serta pemodelan interaksi sistem percakapan [35].

Berdasarkan Gambar 1, metodologi penelitian memadukan keterlacakan artefak rekayasa *Waterfall* pada panel (a) dengan pemodelan hak akses aktor melalui diagram *use case* pada panel (b). Siklus koreksi empiris dari Tahap 4 kembali ke Tahap 2 dan Tahap 3 pada Gambar 1 merepresentasikan penyempurnaan arsitektur berdasarkan temuan pengujian awal, khususnya penambahan jalur deterministik *source-facts* untuk mengatasi halusinasi tarif serta prioritas pencocokan nama terpanjang untuk mengatasi dilusi peringkat semantik. Sementara itu, Gambar 2 memetakan tujuh *use case* utama (UC-01 sampai UC-07) yang memisahkan interaksi Wisatawan (pengguna publik) dari Administrator (pengelola indeks basis pengetahuan).



Gambar 1. Model pengembangan *Waterfall* menurut Sommerville



Gambar 2. Diagram *use case* hak akses aktor Wisatawan dan Administrator.

Identifikasi kebutuhan sistem dilakukan melalui dua sumber: (1) studi pendahuluan dan diskusi kebutuhan informasi bersama pemandu wisata lokal serta pengelola informasi desa wisata pada rangkaian kegiatan penelitian hibah tahun 2026, yang mengidentifikasi empat pertanyaan paling sering diajukan wisatawan (deskripsi daya tarik, jam buka, harga tiket masuk domestik/mancanegara, dan aturan busana adat di pura); serta (2) audit struktur atribut pada korpus 318 dokumen wisata Bali.

Berdasarkan Tabel 1, spesifikasi kebutuhan diformulasikan secara terstruktur menggunakan kode identifikasi unik, tingkat prioritas, rancangan teknis komponen, dan keterlacakan langsung ke kode pengujian (*traceability matrix*). Enam kebutuhan fungsional (KF-01 sampai KF-06) mengatur alur tanya-jawab, akurasi fakta biaya, kartu visual destinasi, manajemen sesi, keamanan administrator, dan pembangunan vektor indeks, sedangkan tiga kebutuhan non-fungsional (KNF-01 sampai KNF-03) menetapkan batasan waktu respons lokal, ketahanan layanan saat model gagal (*graceful degradation*), serta pelindungan privasi data sesi.

Tabel 1. Spesifikasi kebutuhan fungsional, non-fungsional, rancangan teknis, dan keterlacakan pengujian

ID (Prioritas)	Spesifikasi Kebutuhan Sistem	Rancangan Teknis & Keterlacakan Uji
KF-01 (Tinggi)	Menerima kueri bahasa alami (teks/suara) hingga 500 karakter dalam bahasa Indonesia/campuran.	Formulir web, *Web Speech API*, validasi ChatController (UC-01; TC01, TC02, TC04–TC06, NEG02).
KF-02 (Tinggi)	Menyajikan jawaban kontekstual dan tarif tiket yang 100% konsisten terhadap dokumen sumber.	Pencarian hibrida KnowledgeBaseService, llama3.2:3b, & pengarah DestinationFacts (UC-02; TC09, TC11–TC13, GEN-01..03).
KF-03 (Tinggi)	Menampilkan kartu destinasi berisi foto, kabupaten, jam buka, harga selaras, & tautan Google Maps.	Pemetaan *kb_id* ke tabel *places* & unifikasi *price_text* sumber (UC-03; TC11–TC13, DF-03, DF-04).
KF-04 (Sedang)	Menyimpan riwayat sesi percakapan (maks. 10 pesan konteks) & fitur hapus sesi mandiri.	Tabel *chat_messages* terindeks *session_id* & rute POST /chat/clear (UC-04; TC14, TC15).
KF-05 (Tinggi)	Membatasi halaman pengelolaan pengetahuan hanya bagi administrator terautentikasi.	Middleware autentikasi sesi admin & proteksi token CSRF (UC-05; TC03, TC07, TC08).
KF-06 (Tinggi)	Memuat ulang metadata indeks & membangun vektor *embedding* 768 dimensi secara inkremental.	Komando kb:embed dengan pemeriksaan *content_hash* SHA-256 pada *kb_embeddings* (UC-06; TC10).
KNF-01 (Tinggi)	Menghasilkan respons lokal dengan median latensi ≤ 10 detik tanpa ketergantungan API berbayar.	Inferensi lokal Ollama (Q4_K_M) pada Apple M2 16 GB & batas waktu HTTP 30/120 dtk (GEN-04).
KNF-02 (Sedang)	Tetap memberikan jawaban cadangan (*fallback*) ketika layanan model atau indeks vektor tidak aktif.	Cadangan kata kunci berbobot, SQL LIKE wilayah, & pesan *offline* deterministik (NEG01, DF-01).
KNF-03 (Sedang)	Menjaga privasi percakapan secara lokal (anonim) & mencatat telemetri waktu respons serta sumber.	Penyimpanan SQLite lokal *chat_logs* tanpa transmisi pihak ketiga sesuai prinsip UU PDP (UC-07).

2.2 Perancangan Basis Pengetahuan dan Skema Penyimpanan

Korpus basis pengetahuan terdiri atas 318 dokumen berformat *Markdown* (.md) beserta berkas indeks metadata index.json yang dikurasi secara terstruktur oleh tim peneliti. Pengumpulan dan validasi data dilakukan melalui konfirmasi langsung ke pengelola objek wisata di lokasi serta observasi lapangan di Bali, yang dikombinasikan dengan publikasi resmi dinas pariwisata daerah, direktori geospasial OpenStreetMap, dan ensiklopedia publik. Akuisisi data dilaksanakan dalam dua fase: Fase 1 (POI-001 s.d. POI-273; 273 dokumen) dihimpun melalui verifikasi lapangan dan direktori daerah yang dideduplikasi menggunakan skrip normalisasi judul kanonik (deep_deduplicate.py); sedangkan Fase 2 (GMAPS-274 s.d. GMAPS-318; 45 dokumen) ditambahkan untuk melengkapi atribut daya tarik lintas sembilan kabupaten/kota di Bali. Seluruh deskripsi disusun ulang secara parafrasa terstruktur dalam bahasa Indonesia ke dalam empat subbagian baku (Deskripsi, Informasi Lokasi, Operasional & Tiket, serta Etiket & Tips) sehingga bebas dari pelanggaran hak cipta salin-tempel.

Berdasarkan Tabel 2, keseluruhan korpus memuat 37.434 kata (260.920 karakter) dengan rata-rata panjang dokumen 117,7 ± 75,7 kata (820,5 ± 511,0 karakter; median 100,5 kata atau 705,0 karakter; rentang 517 hingga 4.899 karakter). Komposisi kategori didominasi oleh Destinasi Wisata sebanyak 306 dokumen (96,23%), diikuti Kuliner Khas sebanyak 8 dokumen (2,52%), Informasi Praktis 3 dokumen (0,94%), dan Budaya serta Etiket 1 dokumen (0,31%). Audit kelengkapan atribut menunjukkan bahwa 318 dokumen (100%) memiliki informasi lokasi/alamat, 309 dokumen (97,17%) memuat koordinat GPS presisi, dan 273 dokumen (85,85%) mencantumkan jam operasional serta panduan etiket. Untuk atribut harga tiket, sebanyak 114 dokumen (35,85%) memuat angka tarif eksplisit dalam Rupiah (mulai dari Rp 0/Gratis hingga Rp 150.000), 159 dokumen (50,00%) berlabel standar "Gratis / Informasi belum tersedia", dan 45 dokumen tambahan Fase 2 (14,15%) belum mencantumkan baris tiket.

Strategi pemotongan teks (*chunking*) dirancang menggunakan pendekatan satu dokumen utuh = satu vektor *embedding* (318 vektor). Pilihan desain ini didasarkan pada bukti statistik korpus bahwa 98,74% dokumen (314 dari 318 dokumen) memiliki panjang di bawah batas pemotongan 2.000 karakter (~512 token). Karena setiap berkas *Markdown* hanya merepresentasikan satu entitas tunggal yang padat, penyematan tingkat dokumen mempertahankan ikatan semantik antara nama destinasi, kabupaten, jam buka, harga tiket, dan aturan etika dalam satu vektor utuh tanpa risiko terpisahnya atribut dari nama tempat akibat pemotongan sewenang-wenang (*cross-chunk fragmentation*). Penyimpanan persisten diimplementasikan pada basis data relasional SQLite melalui empat tabel terintegrasi: *places*, *kb_embeddings*, *chat_messages*, dan *chat_logs*.

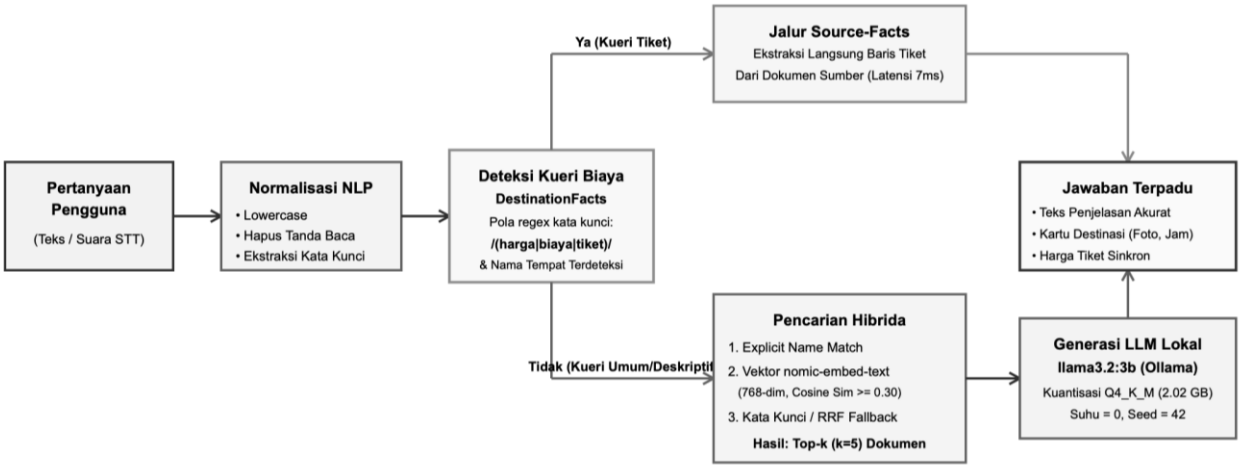
Tabel 2. Statistik korpus basis pengetahuan (318 dokumen) dan kelengkapan atribut

Kategori Dokumen	Jumlah (%)	Panjang (Kata; Karakter)	Kelengkapan (GPS / Jam / Tarif Eksplisit)
Destinasi Wisata	306 (96,23%)	114,3 ± 51,7; 796,9 ± 350,8	303 (99,0%) / 261 (85,3%) / 106 (34,6%)
Kuliner Khas	8 (2,52%)	92,4 ± 7,1; 652,9 ± 42,6	6 (75,0%) / 8 (100%) / 8 (100%)
Informasi Praktis	3 (0,94%)	428,3 ± 336,4; 2.933,7 ± 2.345,4	0 (0,0%) / 3 (100%) / 0 (0,0%)
Budaya & Etiket	1 (0,31%)	440,0 ± 0,0; 3.046,0 ± 0,0	0 (0,0%) / 1 (100%) / 0 (0,0%)
Total / Keseluruhan	318 (100,0%)	117,7 ± 75,7; 820,5 ± 511,0	309 (97,2%) / 273 (85,8%) / 114 (35,8%)

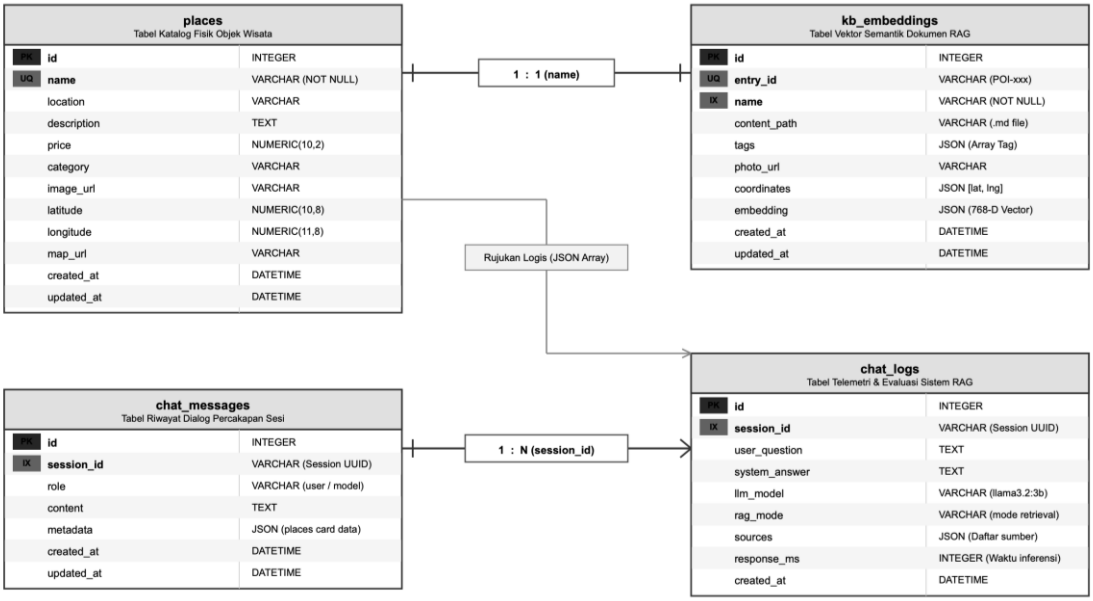
2.3 Perancangan Arsitektur, Alur NLP, dan Reproduksibilitas

Perancangan teknis menyusun lima tahapan pemrosesan bahasa alami (NLP) secara bertingkat untuk membantu model bahasa lokal berukuran 3 miliar parameter agar tetap akurat: (1) pembersihan teks dan pemecahan kata kunci, (2) pencocokan nama tempat wisata dan wilayah di Bali, (3) deteksi pertanyaan biaya berbasis kata kunci, (4) pencarian dokumen gabungan (vektor makna dan kata kunci), serta (5) penyusunan jawaban berdasarkan dokumen sumber.

Gambar 3 memperlihatkan alur pemrosesan bahasa alami (NLP), pencarian dokumen, dan pemisahan jawaban harga tiket, sedangkan Gambar 4 menyajikan hubungan antar-tabel (*Entity Relationship Diagram*/ERD) pada basis data SQLite.



Gambar 3. Alur pemrosesan bahasa alami (NLP), *router* DestinationFacts, dan pencarian dokumen hibrida.



Gambar 4. Skema hubungan antar-tabel (*Entity-Relationship Diagram*).

Berdasarkan Gambar 3, tahap pertama pemrosesan NLP pada KnowledgeBaseService membersihkan kalimat pertanyaan pengguna dengan mengubah huruf besar menjadi huruf kecil (mb_strtolower), menghapus tanda baca (preg_replace), merapikan spasi, dan mengambil kata kunci yang panjangnya ≥ 3 huruf. Tahap kedua menjalankan pencocokan nama tempat pada Jalur 1 (*explicit-name matching*), yaitu memeriksa apakah ada nama dari 318 tempat wisata yang disebut di dalam pertanyaan atau setelah kata depan "di ". Nama tempat yang cocok diurutkan dari yang paling panjang (*longest-specific-match first*) agar nama lengkap seperti "Pura Ulun Danu Beratan" didahulukan daripada nama pendek yang mirip.

Apabila pengguna tidak menyebutkan nama tempat secara langsung, sistem beralih ke Jalur 2 (pencarian vektor makna/*dense semantic retrieval*) menggunakan model *nomic-embed-text:latest* (137 juta parameter, ukuran berkas 274 MB, kode *digest* 0a109f422b47, batas waktu 30 detik). Setiap dokumen diubah menjadi vektor 768 angka dari gabungan judul, daftar tag, dan isi teks (maksimum 2.000 karakter) lalu disimpan pada tabel *kb_embeddings* beserta kode cek SHA-256 (*content_hash*). Saat pertanyaan masuk, tingkat kemiripan makna antara vektor kueri q dan vektor dokumen d dihitung menggunakan metode kemiripan kosinus (*cosine similarity*) sesuai Persamaan (1).

$$\text{sim}(\boldsymbol{q}, \boldsymbol{d}) = \cos(\theta) = \frac{\boldsymbol{q} \cdot \boldsymbol{d}}{\| \boldsymbol{q} \| \ \| \boldsymbol{d} \|} \qquad (1)$$

Dokumen dengan nilai kemiripan ≥ 0,30 diurutkan dari yang tertinggi dan diambil maksimal k = 5 dokumen sebagai konteks masukan model bahasa.

Jika pencarian vektor belum tersedia, sistem menjalankan Jalur 3 (pencarian kata kunci berbobot/*keywordSearch*): kecocokan nama utuh diberi nilai +5, kecocokan kata pada

judul diberi +5, pada tag diberi +3, dan pada isi teks diberi +1, kemudian diambil 5 dokumen dengan nilai tertinggi. Selanjutnya, apabila dokumen yang ditemukan belum terhubung ke kartu tempat wisata pada tabel *places*, sistem menjalankan cadangan wilayah (*regional fallback*) yang mengenali 16 nama kabupaten/kawasan wisata di Bali (Ubud, Gianyar, Badung, Kuta, Seminyak, Nusa Dua, Denpasar, Tabanan, Karangasem, Singaraja, Buleleng, Bangli, Kintamani, Klungkung, Nusa Penida, Jembrana) melalui pencarian SQL LIKE pada kolom *regency*, *name*, dan *address* (maksimal 5 tempat), atau menampilkan 6 tempat wisata populer.

Untuk mencegah model bahasa mengalami halusinasi atau salah menyebutkan angka rupiah, tahap ketiga mendeteksi maksud pertanyaan biaya melalui komponen pengarah fakta destinasi (*DestinationFacts*). Komponen ini secara otomatis mengenali kata kunci terkait biaya seperti harga, biaya, tiket, atau tarif. Apabila pertanyaan terdeteksi menanyakan tarif dan nama tempatnya telah teridentifikasi pada Jalur 1, sistem tidak meminta model bahasa merangkai kalimat baru, melainkan langsung menyajikan informasi tarif resmi dari dokumen sumber (mode*source-facts*) serta menyelaraskan label harga pada kartu destinasi di antarmuka.

Untuk pertanyaan deskriptif dan rekomendasi umum, konteks gabungan (maksimum 5 dokumen) beserta maksimal 10 pesan riwayat sesi terakhir dikirim ke model generasi lokal llama3.2:3b (3,2 miliar parameter, kuantisasi Q4_K_M, ukuran GGUF 2,02 GB, *manifest digest* a80c4f17acd5, batas waktu HTTP 120 detik). *Prompt* sistem didefinisikan secara eksplisit: "Anda adalah Bali *Virtual Tourism Assistant*, asisten wisata virtual yang ramah dan berpengetahuan luas tentang pariwisata Bali. Aturan penting: 1. Selalu jawab dalam Bahasa Indonesia yang baik, sopan, dan ramah. 2. Berikan informasi yang akurat berdasarkan konteks *Knowledge Base* yang diberikan. 3. Sertakan detail praktis seperti estimasi harga tiket, jam buka, lokasi/alamat, dan tips kunjungan jika relevan. 4. Jika ditanya tentang tempat yang tidak ada dalam konteks, katakan secara jujur bahwa Anda belum memiliki informasi detail tentang tempat tersebut, jangan mengarang informasi."

Dari perspektif skema basis data pada Gambar 4 dan kepatuhan pelindungan data pribadi (UU No. 27 Tahun 2022), kolom *kb_id* menjadi kunci penghubung logis antara vektor *kb_embeddings*, kartu *places*, dan larik telemetri *kb_used* pada *chat_logs*. Seluruh riwayat *chat_messages* diisolasi menggunakan pengenal sesi acak anonim (*session_id*) di peladen lokal tanpa mengumpulkan identitas pribadi pengunjung dan dilengkapi tombol penghapusan riwayat sesi oleh pengguna.

2.4 Desain Pengujian dan Evaluasi Empiris

Evaluasi sistem dirancang dalam empat tahapan komplementer: (1) pengujian fungsional *black-box* dan uji batas negatif; (2) pencatatan log defek sebelum dan sesudah perbaikan (*before-vs-after defect log*); (3) studi ablasi dan sensitivitas parameter pencarian konteks; serta (4) evaluasi kualitas generasi berpasangan (*paired generation evaluation*) antara sistem RAG dan *baseline* tanpa RAG.

Seluruh eksperimen dieksekusi pada lingkungan perangkat keras terstandar Apple M2 (arsitektur arm64, CPU 8-inti, memori terpadu 16 GB RAM, sistem operasi macOS, PHP 8.5.3, SQLite 3, dan Ollama versi 0.40.6). Untuk menjamin reproduksibilitas penuh pada evaluasi kualitas jawaban, inferensi model llama3.2:3b dijalankan dengan parameter deterministik *temperature* = 0 dan *seed* = 42, sedangkan latensi diukur dalam satuan milidetik dan dirangkum menggunakan nilai rata-rata, simpangan baku, median, minimum, serta maksimum.

Untuk menguji kemampuan pencarian secara adil, pengujian dibagi menjadi dua kelompok pertanyaan: (a) 954 Pertanyaan Bernama (318 nama tempat saja + 318 permintaan penjelasan bernama + 318 pertanyaan harga tiket bernama) untuk memeriksa apakah seluruh 318 dokumen dapat ditemukan kembali; dan (b) 100 Pertanyaan Variasi Bahasa Bebas yang disusun tanpa menyalin judul dokumen secara langsung, terbagi rata ke dalam lima kelompok (masing-masing n = 20 pertanyaan): S1 Singkatan/Nama Populer (tanpa kata depan seperti "Pura" atau "Pantai"), S2 Deskripsi Tanpa Nama (menyebutkan ciri khas tempat atau aktivitas tanpa menyebut nama tempatnya), S3 Salah Ketik/*Typo* (kesalahan ejaan 1–2 huruf), S4 Bahasa Campuran Indonesia–Inggris (*code-mixing*), dan S5 Kriteria Gabungan Wilayah & Kategori.

Kinerja pencarian diukur menggunakan metrik *Hit@1*, *Hit@5*, *Recall@5*, *Mean Reciprocal Rank* (*MRR@5*), dan *Normalized Discounted Cumulative Gain* (*nDCG@5*), dilengkapi interval kepercayaan proporsi binom Wilson 95% CI. Studi ablasi membandingkan enam konfigurasi pencarian secara terpisah maupun gabungan: (1) aturan nama eksplisit saja, (2) kata kunci berbobot (*keywordSearch*), (3) vektor semantik murni (*nomic-embed-text*, $\tau = 0{,}30$), (4) hibrida bertingkat saat ini (*hard-cutoff cascade*), (5) leksikal BM25 ($k1 = 1{,}5$; $b = 0{,}75$), dan (6) fusi peringkat hibrida *Reciprocal Rank Fusion* (RRF, konstanta $k = 60$). Analisis sensitivitas menguji variasi ambang kemiripan kosinus τ pada rentang {0,20; 0,30; 0,40; 0,50; 0,60; 0,70} dan ukuran konteks k pada {1, 3, 5, 10}.

Konsistensi penyajian fakta biaya dievaluasi secara komparatif sebelum dan sesudah penerapan pengarah DestinationFacts pada enam destinasi pertama dalam indeks (Pura Tanah Lot, Museum Bali Denpasar, Pura Luhur Uluwatu, Garuda Wisnu Kencana, Tegalalang Rice Terrace, dan Sacred Monkey Forest Sanctuary) dengan dua pengulangan (N = 12 pengujian). Suatu respons dinyatakan konsisten apabila seluruh nilai nominal Rupiah pada dokumen *Markdown* sumber muncul secara persis pada teks jawaban, tidak terdapat angka harga tambahan yang tidak

didukung sumber, dan label harga pada kartu destinasi sama dengan dokumen sumber.

Mengacu pada kerangka evaluasi sistem RAG modern [36], [37] serta analisis pemanfaatan konteks oleh model bahasa [38], kualitas jawaban dievaluasi secara berpasangan pada N = 20 pertanyaan uji lintas kategori yang membandingkan keluaran RAG Hibrida (llama3.2:3b + konteks KB + *source-facts*) terhadap *Baseline* Tanpa-RAG (llama3.2:3b murni tanpa pencarian konteks). Setiap respons dinilai menggunakan empat metrik rubrik terukur: (1) *Faithfulness* (proporsi jawaban yang seluruh klaim faktual dan angkanya didukung oleh dokumen sumber tanpa kontradiksi), (2) *Completeness* (proporsi atribut kunci sumber yang berhasil disertakan dalam jawaban), (3) *Answer Relevance* (proporsi jawaban yang menjawab maksud pertanyaan secara langsung), dan (4) *Hallucination Rate* (proporsi jawaban yang memuat klaim lokasi, nama tempat, atau angka tarif fabrikasi).

Seluruh kode sumber aplikasi, skrip pengujian otomatis (run_revisi_mayor.py dan run_generation_eval.py), daftar 100 kueri berstrata, serta rekaman keluaran JSON disimpan secara terstruktur di dalam repositori proyek guna mendukung verifikasi ulang dan pengembangan lanjutan.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Sistem dan Reproduksibilitas Komponen

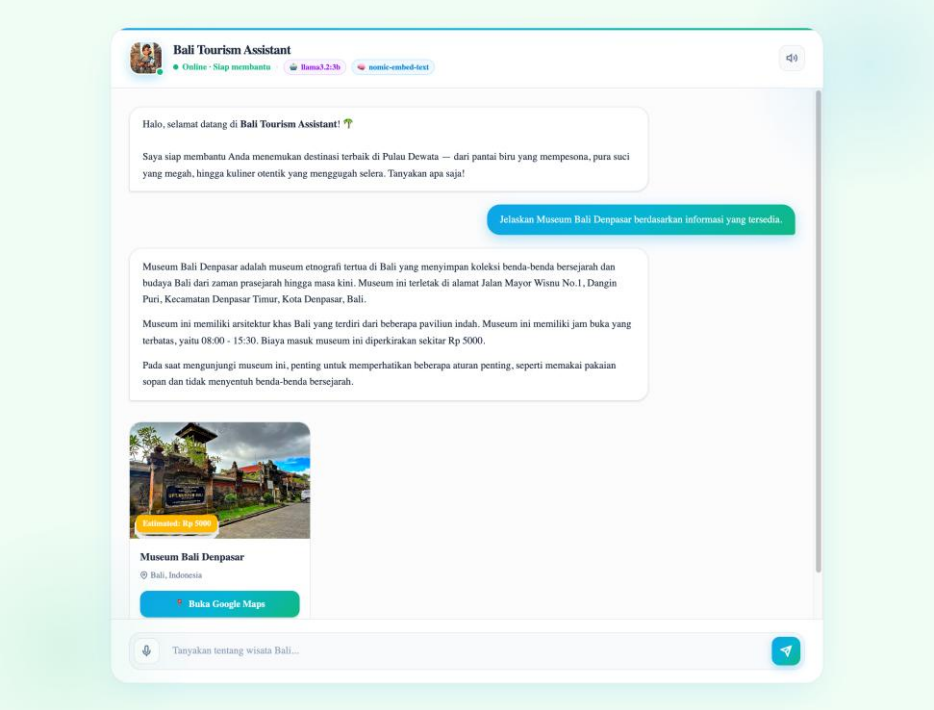
Tahap implementasi menghasilkan prototipe aplikasi web Bali *Virtual Tourism Assistant* dengan arsitektur mandiri *(self-hosted)* yang dapat dijalankan secara fleksibel pada peladen komputasi awan *(cloud server)* maupun server lokal. Spesifikasi perangkat keras pengujian, versi perangkat lunak, identitas *manifest* model bahasa, serta parameter operasional dirangkum pada Tabel 3 untuk menjamin reproduksibilitas sistem.

Tabel 3. Spesifikasi komponen sistem, model bahasa, dan parameter reproduksibilitas

Komponen Sistem	Spesifikasi Teknis, Versi, dan Parameter Reproduksibilitas
Lingkungan Perangkat Keras & OS	Apple M2 (arm64, 8-inti CPU, 16 GB RAM terpadu), macOS, PHP 8.5.3, SQLite 3, Ollama 0.40.6.
Antarmuka & Pengendali Web	Laravel Blade + Tailwind CSS, *Web Speech API* (id-ID), ChatController (validasi maks. 500 karakter, 10 riwayat sesi).
Korpus & Pengindeksan Vektor	318 dokumen *Markdown* + index.json (tervalidasi lapangan); komando kb:embed memproses 318 vektor dalam 28,110 dtk.
Model Embedding Semantik	*nomic-embed-text:latest* (nomic-bert, 137M param, F16, GGUF 274 MB, *digest*: 0a109f422b47, 768 dimensi, $\tau = 0{,}30$, $k = 5$).
Model Generasi Bahasa (LLM)	llama3.2:3b (3,2B param, kuantisasi Q4_K_M, GGUF 2,02 GB, *digest*: a80c4f17acd5, *temperature* = 0, *seed* = 42, timeout 120 dtk).
Pengarah Biaya & Penyimpanan	DestinationFacts (deteksi kata kunci biaya: harga, biaya, tiket, tarif → *source-facts*) & 4 tabel SQLite (*places*, *kb_embeddings*, *chat_messages*, *chat_logs*).

Berdasarkan Tabel 3, proses pembangunan indeks vektor melalui komando artisan kb:embed pada perangkat Apple M2 menyelesaikan penyematan 318 dokumen menjadi 318 vektor berdimensi 768 dalam waktu total 28,110 detik (rata-rata 88,4 milidetik per dokumen) tanpa kegagalan pembacaan berkas. Penggunaan sidik jari SHA-256 pada kolom *content_hash* memastikan bahwa pembaruan indeks berikutnya hanya menghitung ulang dokumen yang mengalami perubahan isi.

Gambar 5 memperlihatkan hasil implementasi antarmuka percakapan saat memproses kueri "Jelaskan Museum Bali Denpasar berdasarkan informasi yang tersedia." Berdasarkan Gambar 5, sistem menyajikan jawaban naratif berbahasa Indonesia yang merangkum deskripsi sejarah, lokasi di Jl. Mayor Wisnu No. 1 Denpasar, jam operasional (08:00–15:30 WITA), serta rincian tarif tiket, disertai kartu visual destinasi di bagian bawah yang menampilkan foto Museum Bali Denpasar secara utuh, label kabupaten/kota, harga tiket yang telah diselaraskan (Rp 5.000), dan tautan navigasi Google Maps.



Gambar 5. Tampilan prototipe pada percakapan mengenai Museum Bali Denpasar.

3.2 Hasil Pengujian Fungsional, Perbaikan *Defect*, dan Kualitas Generasi RAG

Pengujian sistem dilaksanakan secara bertahap untuk memverifikasi kesesuaian fungsional (TC01–TC15), ketahanan terhadap masukan negatif (NEG01–NEG03), efektivitas perbaikan empat *defect* utama yang ditemukan selama siklus pengembangan (DF-01 sampai DF-04), serta peningkatan kualitas jawaban model melalui evaluasi generasi berpasangan (GEN-01 sampai GEN-04).

Rangkuman komprehensif hasil pengujian fungsional, perbandingan metrik sebelum dan sesudah perbaikan *defect*, serta perbandingan kualitas generasi antara konfigurasi RAG Hibrida dan *Baseline* Tanpa-RAG disajikan pada Tabel 4.

Tabel 4. Ringkasan pengujian fungsional, log koreksi *defect*, dan evaluasi generasi berpasangan RAG vs Tanpa-RAG (N = 20)

Kode / Dimensi	Cakupan Skenario dan Metrik Evaluasi	Kondisi Awal / Baseline (Sebelum Koreksi atau Tanpa-RAG)	Kondisi Akhir Sistem (RAG Hibrida + *Source-Facts* + llama3.2:3b)	Status / Capaian
TC01–TC08	Akses halaman utama, aset manifest UI, proteksi CSRF (HTTP 419), validasi pesan kosong/array (HTTP 422), & autentikasi admin.	8 skenario diuji pada antarmuka & pengendali Laravel.	Seluruh 8 skenario sesuai spesifikasi (HTTP 200/302/419/422 tepat).	Lulus (8/8; 100%)
TC09–TC15	Indeks 318 vektor (TC10), generasi naratif (TC09, TC11, TC12), pengarah biaya (TC13), penyimpanan & hapus sesi (TC14, TC15).	TC09 sempat gagal pada konfigurasi awal sebelum penyelarasan model.	7 skenario lulus; indeks 318 vektor selesai 28,11 dtk; sesi terisolasi.	Lulus (7/7; 100%)
NEG01–NEG03	Uji negatif: ketiadaan model (llama3.1), pesan >500 karakter, & pertanyaan jebakan di luar domain Bali.	Tanpa penanganan kesalahan, ketiadaan model memicu galat HTTP 500.	NEG01 beralih ke *fallback offline* (HTTP 200); NEG02 ditolak (HTTP 422).	Tertangani (3/3)
DF-01 (Defek Model)	Ketidakcocokan nama model default (llama3.1) terhadap model terpasang pada layanan Ollama lokal.	Gagal HTTP 500 ("model llama3.1 not found") pada pengujian awal TC09.	Konfigurasi diselaraskan ke llama3.2:3b (Q4_K_M) + *fallback* otomatis.	Terselesaikan
DF-02 (Dilusi Vektor)	Dilusi kata umum ("berapa harga tiket masuk") pada pencarian vektor murni untuk 954 kueri bernama.	Vektor murni: *Hit@1* = 78,93% (753/954); kueri tiket *Hit@1* = 60,38%.	Prioritas nama spesifik terpanjang (*explicit-name*): *Hit@1* = 100% (954/954).	+21,07 poin *Hit@1*

Kode / Dimensi	Cakupan Skenario dan Metrik Evaluasi	Kondisi Awal / Baseline (Sebelum Koreksi atau Tanpa-RAG)	Kondisi Akhir Sistem (RAG Hibrida + *Source-Facts* + llama3.2:3b)	Status / Capaian
DF-03 (Halusinasi Tarif)	Konsistensi angka tarif pada teks jawaban & kartu destinasi (12 uji tiket pada 6 destinasi utama).	Generasi LLM murni: hanya 1/12 (8,33%) akurat; 6/12 harga kartu konflik.	Pengarah fakta DestinationFacts + sinkronisasi harga kartu: 12/12 (100,00%) akurat.	+91,67 poin akurasi
DF-04 (Aset Gambar)	Keterhubungan tautan simbolik gambar destinasi pada kartu antarmuka percakapan.	Tautan absolut lama terputus sehingga gambar kartu gagal dimuat di UI.	Perbaikan symlink relatif public/*kb_images* ke korpus: 318/318 gambar aktif.	Terselesaikan
GEN-01 (*Faithfulness*)	Kesetiaan klaim faktual & angka jawaban terhadap dokumen sumber pada N = 20 kueri berpasangan.	*Baseline* Tanpa-RAG: 2/20 (10,00%; Wilson 95% CI: 2,79%–30,10%).	RAG Hibrida: 18/20 (90,00%; Wilson 95% CI: 69,90%–97,21%).	+80,00 poin
GEN-02 (*Completeness*)	Proporsi fakta kunci sumber (lokasi, jam, tarif, aturan) yang tercakup dalam jawaban (N = 20).	*Baseline* Tanpa-RAG: rata-rata kelengkapan fakta 23,60%.	RAG Hibrida: rata-rata kelengkapan fakta mencapai 92,00%.	+68,40 poin
GEN-03 (Halusinasi)	Relevansi jawaban terhadap maksud kueri & tingkat halusinasi faktual (N = 20).	Tanpa-RAG: Relevansi 75,00% (15/20); Halusinasi 90,00% (18/20).	RAG Hibrida: Relevansi 95,00% (19/20); Halusinasi turun ke 10,00% (2/20).	-80,00 poin halusinasi
GEN-04 (Latensi)	Distribusi waktu respons inferensi pada Apple M2 16 GB (*temperature* = 0, *seed* = 42).	Tanpa-RAG (n = 20): rerata 5,96 ± 2,55 dtk (median 7,22; rentang 1,92–8,96).	RAG LLM (n = 18): 7,28 ± 4,83 dtk (median 6,50); *Source-Facts* (n = 2): 0,007 dtk.	Median 6,50 dtk

Berdasarkan Tabel 4, seluruh 15 kasus uji fungsional (TC01–TC15) dan 3 uji negatif (NEG01–NEG03) berhasil dipenuhi (100%). Log perbaikan *defect* mendokumentasikan secara transparan bagaimana siklus verifikasi-koreksi mengatasi kelemahan awal: sebelum penerapan DestinationFacts (DF-03), pengujian 12 kueri biaya pada model llama3.2:3b murni hanya menghasilkan 1 dari 12 jawaban (8,33%) yang memuat seluruh angka sumber dengan benar; 3 jawaban mengarang tarif yang tidak ada di sumber (seperti Rp 30.000 atau Rp 80.000), 7 jawaban gagal menyalin tarif karena terdistraksi oleh dokumen lain di dalam konteks *top-5*, dan 6 kartu menampilkan harga berbeda dari teks (misalnya "Gratis" pada kartu Tegalalang dan GWK). Setelah pengarah regex *source-facts* dan penyelarasan harga kartu diaktifkan, tingkat kesesuaian biaya meningkat menjadi 12/12 (100,00%; Wilson 95% CI: 75,75%–100,00%) dengan waktu eksekusi rata-rata 7 milidetik. Lebih lanjut, evaluasi berpasangan pada N = 20 kueri menunjukkan bahwa arsitektur RAG hibrida meningkatkan *Faithfulness* sebesar +80,00 poin persentase (dari 10,00% menjadi 90,00%) dan *Completeness* sebesar +68,40 poin persentase (dari 23,60% menjadi 92,00%) dibanding llama3.2:3b tanpa RAG, dengan median waktu respons 6,50 detik (rata-rata 7,28 ± 4,83 detik).

3.3 Evaluasi Pencarian Konteks, Perbandingan Metode, dan Sensitivitas Parameter

Untuk menguji kontribusi komponen pencarian secara objektif, evaluasi dilakukan baik pada 954 pertanyaan bernama maupun pada 100 pertanyaan variasi bahasa bebas (kelompok S1–S5). Hasil perbandingan antar-metode pencarian, kinerja pada setiap kelompok bahasa, serta pengujian nilai ambang kemiripan kosinus (τ) dan jumlah dokumen konteks (k) disajikan pada Tabel 5.

Tabel 5. Hasil perbandingan metode pencarian (954 pertanyaan bernama dan 100 pertanyaan variasi bahasa S1–S5) serta uji parameter

Metode Pencarian / Kelompok Uji	Hit@1 (95% CI)	Hit@5 (95% CI)	Recall@5 / MRR@5 / nDCG@5
Uji 954 Bernama: Vektor Semantik Murni	78,93% (76,2–81,4%)	91,82% (89,9–93,4%)	0,9182 / 0,8387 / 0,8586
Uji 954 Bernama: Kata Kunci Berbobot	97,06% (95,8–98,0%)	99,27% (98,5–99,6%)	0,9927 / 0,9803 / 0,9835
Uji 954 Bernama: Gabungan Sistem (Nama)	100,0% (99,6–100%)	100,0% (99,6–100%)	1,0000 / 1,0000 / 1,0000
Uji 100 Bebas: Pencocokan Nama Saja	21,00% (14,2–30,0%)	21,00% (14,2–30,0%)	0,1973 / 0,2100 / 0,2008
Uji 100 Bebas: Kata Kunci Berbobot	66,00% (56,3–74,5%)	73,00% (63,6–80,7%)	0,6823 / 0,6887 / 0,6652
Uji 100 Bebas: Vektor Makna (nomic, τ=0,30)	66,00% (56,3–74,5%)	85,00% (76,7–90,7%)	0,8010 / 0,7298 / 0,7283

Metode Pencarian / Kelompok Uji	Hit@1 (95% CI)	Hit@5 (95% CI)	Recall@5 / MRR@5 / nDCG@5
Uji 100 Bebas: Bertingkat Saat Ini	62,00% (52,2–70,9%)	79,00% (70,0–85,8%)	0,7407 / 0,6848 / 0,6846
Uji 100 Bebas: Kata Kunci BM25	86,00% (77,9–91,5%)	96,00% (90,2–98,4%)	0,9263 / 0,9007 / 0,8920
Uji 100 Bebas: Gabungan Peringkat (RRF)	75,00% (65,7–82,5%)	96,00% (90,2–98,4%)	0,9197 / 0,8387 / 0,8420
Kelompok S1 Nama Singkat (Vektor / RRF)	85,00% / 85,00%	95,00% / 100,0%	0,9500 / 0,8875 / 0,9031 (Vek)
Kelompok S2 Tanpa Nama (Vektor / RRF)	45,00% / 65,00%	70,00% / 95,00%	0,7000 / 0,5367 / 0,5769 (Vek)
Kelompok S3 Salah Ketik (Vektor / RRF)	70,00% / 75,00%	85,00% / 90,00%	0,8500 / 0,7492 / 0,7734 (Vek)
Kelompok S4 Campuran ID-EN (Vektor/RRF)	55,00% / 75,00%	75,00% / 95,00%	0,7500 / 0,6375 / 0,6656 (Vek)
Kelompok S5 Kriteria Wilayah (Vektor/RRF)	75,00% / 75,00%	100,0% / 100,0%	0,7550 / 0,8383 / 0,7225 (Vek)
Uji Ambang τ = 0,20..0,50 (k=5)	66,00% (stabil)	85,00% (5,0 dok)	*Recall@5* = 0,8010; *MRR@5* = 0,7298
Uji Ambang τ = 0,60 / τ = 0,70	66,00% / 47,00%	85,00% / 53,00%	Rerata dok: 4,67 (0,60) / 1,51 (0,70)
Uji Jumlah Dokumen k=1 / k=3 / k=5 / k=10	66,00% (konstan)	66% / 79% / 85% / 88%	Recall@k: 0,530 / 0,714 / 0,801 / 0,844

Berdasarkan Tabel 5, perbandingan metode menunjukkan bahwa setiap jalur pencarian saling melengkapi. Pada 954 pertanyaan bernama, pencarian vektor *nomic-embed-text* murni menghasilkan *Hit@1* sebesar 78,93% dan *Hit@5* sebesar 91,82% karena kalimat tanya yang mirip pada pertanyaan tiket ("Berapa harga tiket masuk...") kadang mengangkat dokumen pura lain ke urutan pertama, sehingga penambahan aturan pencocokan nama diperlukan agar *Hit@1* mencapai 100% saat nama tempat ditulis lengkap. Sebaliknya, pada 100 pertanyaan variasi bahasa bebas (S1–S5), aturan pencocokan nama saja hanya mampu menemukan 21,00% tempat wisata (0% pada S2 Deskripsi Tanpa Nama, S3 Salah Ketik, dan S4 Bahasa Campuran). Pada pertanyaan bebas ini, pencarian vektor makna *nomic-embed-text* mencapai *Hit@5* sebesar 85,00% (*Recall@5* = 0,8010; *MRR@5* = 0,7298), mengungguli pencarian kata kunci berbobot (*Hit@5* = 73,00%) sebesar +12,00 poin, terutama pada S2 Deskripsi Tanpa Nama yang naik dua kali lipat (*Hit@5* 70,00% vs. 35,00%) serta S4 Bahasa Campuran Indonesia–Inggris (*Hit@5* 75,00% vs. 50,00%).

Pengujian parameter pada Tabel 5 juga menjelaskan alasan pemilihan angka pengaturan pada aplikasi. Ambang kemiripan kosinus τ = 0,30 berada pada rentang yang aman (nilai τ antara 0,20 sampai 0,50 menjaga *Hit@5* tetap 85,00% dengan rata-rata 5 dokumen), sedangkan jika ambang dinaikkan terlalu tinggi ke τ = 0,70, banyak dokumen yang terbuang sehingga *Hit@5* turun menjadi 53,00%. Untuk jumlah dokumen konteks k, pemilihan k = 5 memberikan keseimbangan terbaik (*Recall@5* = 0,8010 dibanding k = 1 sebesar 0,5303 dan k = 3 sebesar 0,7140), sedangkan penambahan menjadi k = 10 hanya menambah sedikit ketepatan (*Recall@10* = 0,8443) tetapi membuat teks konteks terlalu panjang bagi model 3B [38].

3.4 Analisis Kasus Kegagalan dan Perbandingan Penelitian

Pemeriksaan terhadap kasus yang belum tepat pada Tabel 5 menunjukkan temuan menarik: alur pencarian bertingkat saat ini mencatat *Hit@5* sebesar 79,00% pada 100 pertanyaan bebas, sedikit di bawah pencarian vektor murni (85,00%). Hal ini terjadi pada 6 pertanyaan yang menyebut nama lokasi umum yang kebetulan juga terdaftar sebagai judul dokumen tersendiri. Contohnya pada pertanyaan S1-05 ("tempat makan sup ikan bakar legendaris Mak Beng di dekat Pantai Sanur") dan S2-01 ("pura terbesar di Bali yang terletak di lereng Gunung Agung"), Jalur 1 langsung mengambil dokumen "Pantai Sanur" (POI-250) dan "Gunung Agung" (POI-091) lalu berhenti mencari dokumen tujuan utamanya ("Warung Mak Beng" [POI-305] dan "Pura Agung Besakih" [POI-181]). Ketika hasil pencocokan nama, kata kunci BM25, dan vektor makna digabungkan peringkatnya menggunakan *Reciprocal Rank Fusion* (RRF), kelemahan tersebut teratasi dan *Hit@5* meningkat menjadi 96,00% (*MRR@5* = 0,8387).

Pada pengujian kualitas jawaban untuk 20 pertanyaan berpasangan, terlihat perbedaan jelas antara sistem yang diberi dokumen sumber (RAG) dan model AI tanpa dokumen (Tanpa-RAG). Tanpa dokumen sumber, model llama3.2:3b memberikan informasi yang salah pada 18 dari 20 pertanyaan (90,00%): misalnya pada pertanyaan G08, model menyebut Kelingking Beach (yang berada di Nusa Penida, Klungkung) terletak di Kintamani, Bangli; pada G14, model menyebut Diamond Beach berada di Kepulauan Cook dan memiliki es gletser; serta pada G11 mengarang nama "Pura Batukandikusumah". Dengan sistem RAG yang dibangun, 18 dari 20 jawaban (90,00%) sudah sesuai dengan isi dokumen. Dua jawaban yang belum sempurna berasal dari: (1) pertanyaan G13 (Pura Goa Giri Putri), di mana model sudah benar menjelaskan aturan kain kamen tetapi menambahkan kalimat "penggunaan kamera dilarang" yang tidak tertulis di dokumen; serta (2) pertanyaan uji di luar wilayah Bali G17 (Gunung Bromo di Jawa Timur), di mana kata "gunung" masih menarik dokumen gunung di Bali

sehingga model tetap menjawab. Selain itu, pada G10 (Desa Penglipuran), model sudah tepat menyebutkan tiket Rp 30.000 tetapi belum mencantumkan jam buka 08:00–17:00 WITA.

Pada pencarian vektor murni untuk kelompok S2 (Deskripsi Tanpa Nama), masih ada 6 dari 20 pertanyaan yang belum masuk 5 besar, seperti S2-09 ("tempat melihat patung raksasa Dewa Wisnu mengendarai burung Garuda di bukit kapur Jimbaran" untuk GWK) dan S2-13 ("desa wisata terkenal yang sangat bersih tanpa kendaraan bermotor di Bangli" untuk Desa Penglipuran). Hal ini terjadi karena kalimat di dokumen menggunakan istilah seperti "taman budaya megah" dan "konsep Tri Hita Karana", sedangkan model *nomic-embed-text* yang dilatih dominan dalam bahasa Inggris belum selalu mengenali padanan kalimat sehari-hari bahasa Indonesia tanpa bantuan pencarian kata kunci.

Dibandingkan dengan penelitian *chatbot* pariwisata sebelumnya, hasil ini memberikan bukti perbaikan yang nyata. Berbeda dari penelitian Handayani dkk. [12], Cruz dkk. [13], dan Sperlí [14] yang belum memisahkan penanganan angka harga tiket pada model generatif lokal, sistem ini menunjukkan bahwa pemisahan pertanyaan biaya ke jalur pembacaan dokumen langsung (*source-facts*) mampu meningkatkan ketepatan informasi harga tiket dari 8,33% (1/12) menjadi 100% (12/12 tepat).

Dibandingkan pula dengan penelitian RAG lokal oleh Albert dan Voutama [31] yang menguji satu dokumen PDF dengan Ollama tanpa membandingkan metode pencarian maupun menguji model tanpa RAG, penelitian ini membuktikan pada 318 dokumen wisata bahwa penggabungan pencocokan nama, pencarian vektor, dan pembacaan fakta tiket mampu meningkatkan ketepatan fakta jawaban dari 10,00% menjadi 90,00% dengan waktu tunggu median 6,50 detik.

3.5 Manfaat Praktis dan Etika Penyampaian Informasi Wisata

Hasil pengujian menunjukkan bahwa kekurangan model bahasa berbobot ringan berukuran 3B (llama3.2:3b) dan model vektor bahasa Inggris (*nomic-embed-text*) saat memproses pertanyaan bahasa Indonesia [23], [29], [30] dapat diatasi dengan menggabungkan pencocokan nama tempat, pencarian kata kunci, dan pembacaan atribut tiket secara langsung. Dari sisi manfaat praktis, rancangan aplikasi ini menerapkan prinsip operasional tanpa biaya langganan API berbayar per token (*zero API cost*) sehingga fleksibel untuk dijalankan secara mandiri, baik pada infrastruktur peladen komputasi awan (*cloud server* / VPS) maupun server lokal berkapasitas RAM 16 GB oleh dinas pariwisata atau pengelola desa wisata.

Dari perspektif sosial-budaya dan kebijakan pariwisata berkelanjutan di Bali, perancangan sistem ini memiliki implikasi krusial dalam memitigasi maraknya insiden pelanggaran etika dan norma kesucian di kawasan pura oleh wisatawan, khususnya wisatawan mancanegara. Berbagai insiden faktual di lapangan, seperti wisatawan yang menaiki atau memanjat bangunan suci pelinggih dan padmasana demi konten media sosial, berbusana tidak pantas di area persembahyangan, hingga memasuki mandala utama jeroan pura dalam kondisi cuntaka (seperti masa berkabung atau datang bulan) tanpa mengenakan wastra kepatutan adat (kain kamen dan selendang), telah menimbulkan keprihatinan mendalam masyarakat adat serta melahirkan regulasi ketat seperti Surat Edaran Gubernur Bali Nomor 4 Tahun 2023 tentang Tatanan Baru bagi Wisatawan Mancanegara selama Berada di Bali (pedoman *Do's and Don'ts*). Model LLM generatif murni tanpa penambatan fakta lokal terbukti berisiko tinggi karena kerap mengabaikan larangan sakral tersebut atau menurunkannya menjadi sekadar anjuran opsional. Melalui penambatan fakta aturan secara deterministik (atribut *rules*, *dress_code*, dan *cultural_notes*), Bali Virtual Tourism Assistant berfungsi sebagai pagar pengaman budaya digital preventif (*preventive digital cultural guardrails*) yang secara otomatis menyuntikkan panduan etiket sakral setiap kali kueri mendeteksi objek pura, sehingga mengedukasi wisatawan sejak fase sebelum kedatangan (*pre-trip phase*) dan meminimalkan risiko penistaan kawasan suci.

Dari perspektif tata kelola data operasional dan kepatuhan hukum, mengingat tarif tiket masuk, jam operasional, serta jadwal upacara keagamaan (piodalan) di pura dapat berubah secara dinamis, data kurasi dokumen per Oktober 2026 perlu diperbarui secara berkala oleh administrator pengelola melalui antarmuka web yang terintegrasi. Di samping itu, penyimpanan riwayat percakapan yang diisolasi pada peladen mandiri (*self-hosted server*), baik berbasis cloud privat maupun lokal, menggunakan identifikasi sesi acak menjamin kepatuhan penuh terhadap prinsip pelindungan data pribadi pengguna sesuai amanat Undang-Undang Republik Indonesia Nomor 27 Tahun 2022 tentang Pelindungan Data Pribadi (UU PDP).

3.6 Keterbatasan Penelitian dan Pengembangan Lanjutan

Penelitian ini masih memiliki beberapa keterbatasan. Pertama, isi basis pengetahuan yang berjumlah 318 dokumen masih didominasi oleh tempat wisata (306 dokumen atau 96,23%), sedangkan dokumen kuliner (8 dokumen), informasi praktis (3 dokumen), dan budaya umum (1 dokumen) masih sedikit. Kedua, alur pencarian bertingkat saat ini langsung berhenti ketika menemukan nama tempat di dalam kalimat pertanyaan, sehingga ke depan perlu diganti dengan metode gabungan peringkat (RRF). Ketiga, aplikasi belum memiliki penyaring otomatis untuk menolak pertanyaan wisata di luar Provinsi Bali (seperti pertanyaan tentang Gunung Bromo).

Keempat, pengujian dalam penelitian ini baru mencakup uji fungsi sistem, perbandingan metode pencarian, dan

pemeriksaan ketepatan isi jawaban terhadap dokumen; penelitian ini belum melibatkan kuesioner kepuasan wisatawan secara langsung menggunakan *System Usability Scale* (SUS) atau *User Experience Questionnaire* (UEQ), uji beban banyak pengguna sekaligus, maupun uji akurasi fitur suara di berbagai peramban. Pengembangan selanjutnya akan difokuskan pada penambahan dokumen kuliner dan budaya Bali, penerapan pencarian gabungan RRF dengan model vektor multibahasa, serta pengujian langsung bersama wisatawan.

4. KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian empiris, disimpulkan bahwa: (1) Dari aspek perancangan sistem, arsitektur Bali *Virtual Tourism Assistant* yang menghubungkan aplikasi Laravel, basis pengetahuan 318 dokumen *Markdown*, vektor 768 dimensi *nomic-embed-text,* pengarah fakta biaya DestinationFacts, dan model bahasa llama3.2:3b berhasil memenuhi seluruh spesifikasi kebutuhan dengan kelulusan uji fungsional *black-box* 100% (15/15) serta penanganan penuh tiga skenario negatif. (2) Dari aspek kinerja pencarian konteks, pada 954 pertanyaan bernama aturan prioritas nama terpanjang meningkatkan *Hit@1* dari 78,93% menjadi 100,00%, sedangkan pada 100 pertanyaan variasi bahasa bebas (S1–S5) pencarian vektor semantik mencapai *Hit@5* sebesar 85,00% (*Recall@5* = 0,8010; *MRR@5* = 0,7298) pada pengaturan $\tau = 0{,}30$ dan $k = 5$ yang mengungguli kata kunci berbobot (*Hit@5* = 73,00%) terutama pada deskripsi tanpa nama (70,00% vs. 35,00%) dan bahasa campuran (75,00% vs. 50,00%), serta dapat ditingkatkan hingga *Hit@5* = 96,00% melalui penggabungan peringkat (RRF).

(3) Dari aspek kualitas generasi jawaban, evaluasi berpasangan pada 20 pertanyaan membuktikan bahwa integrasi RAG berhasil meningkatkan ketepatan fakta jawaban (*faithfulness*) dari 10,00% (2/20 pada model llama3.2:3b tanpa dokumen sumber) menjadi 90,00% (18/20), meningkatkan kelengkapan isi dari 23,60% menjadi 92,00%, dan menurunkan kesalahan informasi dari 90,00% menjadi 10,00% dengan median waktu respons 6,50 detik. Selain itu, pengalihan pertanyaan harga tiket ke pembacaan dokumen langsung (*source-facts*, waktu respons 7 milidetik) dan penyamaan harga pada kartu destinasi berhasil meningkatkan ketepatan informasi biaya dari 8,33% (1/12) menjadi 100,00% (12/12 tepat).

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Direktorat Penelitian dan Pengabdian kepada Masyarakat, Direktorat Jenderal Riset dan Pengembangan, Kementerian Pendidikan Tinggi, Sains, dan Teknologi Republik Indonesia atas pendanaan Penelitian Dosen Pemula (PDP) Tahun Anggaran 2026, serta kepada Universitas Markandeya Bali dan Institut Teknologi dan Bisnis STIKOM Bali atas dukungan fasilitas penelitian.

DAFTAR PUSTAKA

[1] T. Pencarelli, “The digital revolution in the travel and tourism industry,” *Inf. Technol. Tourism*, vol. 22, no. 3, pp. 455–476, 2020, doi: 10.1007/s40558-019-00160-3.

[2] M. Sigala et al., “Understanding the impact of ChatGPT on tourism and hospitality: Trends, prospects and research agenda,” *J. Hospitality Tourism Manage.*, vol. 60, pp. 384–390, 2024, doi: 10.1016/j.jhtm.2024.08.004.

[3] Y. Li, C. Hu, C. Huang, and L. Duan, “The concept of smart tourism in the context of tourism information services,” *Tourism Manage.*, vol. 58, pp. 293–300, 2017, doi: 10.1016/j.tourman.2016.03.014.

[4] D. C. Ukpabi and H. Karjaluoto, “Consumers’ acceptance of information and communications technology in tourism: A review,” *Telematics Informat.*, vol. 34, no. 5, pp. 618–644, 2017, doi: 10.1016/j.tele.2016.12.002.

[5] R. Pillai and B. Sivathanu, “Adoption of AI-based chatbots for hospitality and tourism,” *Int. J. Contemp. Hospitality Manage.*, vol. 32, no. 10, pp. 3199–3226, 2020, doi: 10.1108/ijchm-04-2020-0259.

[6] E. Adamopoulou and L. Moussiades, “Chatbots: History, technology, and applications,” *Mach. Learn. Appl.*, vol. 2, Art. no. 100006, 2020, doi: 10.1016/j.mlwa.2020.100006.

[7] A. Casheekar, A. Lahiri, K. Rath, K. S. Prabhakar, and K. Srinivasan, “A contemporary review on chatbots, AI-powered virtual conversational agents, ChatGPT: Applications, open challenges and future research directions,” *Comput. Sci. Rev.*, vol. 52, Art. no. 100632, 2024, doi: 10.1016/j.cosrev.2024.100632.

[8] A. Følstad et al., “Future directions for chatbot research: An interdisciplinary research agenda,” *Computing*, vol. 103, no. 12, pp. 2915–2942, 2021, doi: 10.1007/s00607-021-01016-7.

[9] B. E. Bakkouri, S. Raki, and T. Belgnaoui, “The role of chatbots in enhancing customer experience: Literature review,” *Procedia Comput. Sci.*, vol. 203, pp. 432–437, 2022, doi: 10.1016/j.procs.2022.07.057.

[10] A. Følstad and P. B. Brandtzaeg, “Users’ experiences with chatbots: Findings from a questionnaire study,” *Qual. User Exp.*, vol. 5, no. 1, Art. no. 3, 2020, doi: 10.1007/s41233-020-00033-2.

[11] M. R. Suherlan, A. Asriyanik, and A. Pambudi, “UMMIBOT sebagai media layanan informasi penerimaan mahasiswa baru Universitas Muhammadiyah Sukabumi,” *Jurnal Informatika Terpadu*, vol. 9, no. 2, pp. 82–91, 2023, doi: 10.54914/jit.v9i2.893.

[12] S. F. Handayani, Dairoh, and D. I. Af'idah, "Assessment of retrieval and generative chatbots in tourism information service," *JUITA: Jurnal Informatika*, vol. 13, no. 1, pp. 19–27, 2025, doi: 10.30595/juita.v13i1.24182.

[13] M. Cruz, B. Jardim, and M. D. C. Neto, "Lisa: A touristic chatbot for Lisbon," *Inf. Technol. Tourism*, vol. 27, no. 4, pp. 1153–1183, 2025, doi: 10.1007/s40558-025-00339-x.

[14] G. Sperlí, "A cultural heritage framework using a deep learning based chatbot for supporting tourist journey," *Expert Syst. Appl.*, vol. 183, Art. no. 115277, 2021, doi: 10.1016/j.eswa.2021.115277.

[15] L. Benaddi, C. Ouaddi, A. Jakimi, and B. Ouchao, "Towards a software factory for developing the chatbots in smart tourism mobile applications," *Procedia Comput. Sci.*, vol. 231, pp. 275–280, 2024, doi: 10.1016/j.procs.2023.12.203.

[16] A. Prasanna, P. Pushparaj, and B. P. Kushwaha, "Conversational AI in tourism: A systematic literature review using TCM and ADO framework," *J. Hospitality Tourism Manage.*, vol. 64, Art. no. 101310, 2025, doi: 10.1016/j.jhtm.2025.101310.

[17] Q. D. Tuyen, T. Vo-Thanh, N. H. Nghi, B. T. Phuong, and B. T. T. Bao, "Research on ChatGPT in tourism and hospitality: A systematic review and future directions," *Int. J. Hospitality Manage.*, vol. 131, Art. no. 104287, 2025, doi: 10.1016/j.ijhm.2025.104287.

[18] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems (NeurIPS)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., Red Hook, NY, USA: Curran Associates, Inc., vol. 33, pp. 9459–9474, 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

[19] P. Zhao et al., "Retrieval-augmented generation for AI-generated content: A survey," *Data Sci. Eng.*, vol. 11, no. 1, pp. 1–29, 2026, doi: 10.1007/s41019-025-00335-5.

[20] Z. Ji et al., "Survey of hallucination in natural language generation," *ACM Comput. Surv.*, vol. 55, no. 12, Art. no. 248, 2023, doi: 10.1145/3571730.

[21] L. Huang et al., "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," *ACM Trans. Inf. Syst.*, vol. 43, no. 2, Art. no. 42, 2025, doi: 10.1145/3703155.

[22] A. Brown, M. Roman, and B. Devereux, "A systematic literature review of retrieval-augmented generation: Techniques, metrics, and challenges," *Big Data Cogn. Comput.*, vol. 9, no. 12, Art. no. 320, 2025, doi: 10.3390/bdcc9120320.

[23] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking large language models in retrieval-augmented generation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 16, pp. 17754–17762, Mar. 2024, doi: 10.1609/aaai.v38i16.29728.

[24] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, Hong Kong, China, Nov. 2019, pp. 3982–3992, doi: 10.18653/v1/D19-1410.

[25] J. Camacho-Collados and M. T. Pilehvar, "From word to sense embeddings: A survey on vector representations of meaning," *J. Artif. Intell. Res.*, vol. 63, pp. 743–788, 2018, doi: 10.1613/jair.1.11259.

[26] V. Karpukhin et al., "Dense passage retrieval for open-domain question answering," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Nov. 2020, pp. 6769–6781, doi: 10.18653/v1/2020.emnlp-main.550.

[27] Z. Nussbaum, J. X. Morris, B. Duderstadt, and A. Mulyar, "Nomic Embed: Training a reproducible long context text embedder," *Trans. Mach. Learn. Res.*, ISSN 2835-8856, Feb. 2025. [Online]. Available: https://openreview.net/forum?id=IPmzyQSiQE

[28] J. Chen et al., "M3-Embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation," in *Findings Assoc. Comput. Linguistics: ACL 2024*, Bangkok, Thailand, Aug. 2024, pp. 2318–2335, doi: 10.18653/v1/2024.findings-acl.137.

[29] B. Wilie et al., "IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding," in *Proc. 1st Conf. Asia-Pacific Chapter Assoc. Comput. Linguistics 10th Int. Joint Conf. Natural Lang. Process. (AACL-IJCNLP)*, Suzhou, China, Dec. 2020, pp. 843–857, doi: 10.18653/v1/2020.aacl-main.85.

[30] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A benchmark dataset and pre-trained language model for Indonesian NLP," in *Proc. 28th Int. Conf. Comput. Linguistics (COLING)*, Barcelona, Spain, Dec. 2020, pp. 757–770, doi: 10.18653/v1/2020.coling-main.66.

[31] M. Saman, G. A. F. Alfarisy, R. Amelia, and N. R. Fadhliana, "A RAG-based academic information chatbot using lightweight LLaMA and Indo-Sentence-BERT," Indones. J. Artif. Intell. Data Mining, vol. 8, no. 3, pp. 687–696, 2025, doi: 10.24014/ijaidm.v8i3.38150.

[32] T. Taipalus, "Vector database management systems: Fundamental concepts, use-cases, and current challenges," *Cogn. Syst. Res.*, vol. 85, Art. no. 101216, 2024, doi: 10.1016/j.cogsys.2024.101216.

[33] I. Sommerville, *Software Engineering*, 10th ed., Global ed. Harlow, U.K.: Pearson, 2016, ISBN: 978-1-292-09613-1.

[34] S. Amershi et al., "Software engineering for machine learning: A case study," in *Proc. IEEE/ACM 41st Int. Conf. Softw. Eng.: Softw. Eng. Pract. (ICSE-SEIP)*, Montreal, QC, Canada, May 2019, pp. 291–300, doi: 10.1109/ICSE-SEIP.2019.00042.

[35] M. R. Herfian and A. R. Adriansyah, "Analisis dan perancangan aplikasi chatbot dalam pelayanan penerimaan mahasiswa baru pada perguruan tinggi," *Jurnal Informatika Terpadu*, vol. 7, no. 2, pp. 87–93, 2021, doi: 10.54914/jit.v7i2.370.

[36] S. Es, J. James, L. Espinosa Anke, and S. Schockaert, "RAGAs: Automated evaluation of retrieval augmented generation," in *Proc. 18th Conf. Eur. Chapter Assoc. Comput. Linguistics: Syst. Demonstrations (EACL)*, St. Julian's, Malta, Mar. 2024, pp. 150–158, doi: 10.18653/v1/2024.eacl-demo.16.

[37] J. Saad-Falcon, O. Khattab, C. Potts, and M. Zaharia, "ARES: An automated evaluation framework for retrieval-augmented generation systems," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol. (NAACL-HLT)*, vol. 1, Mexico City, Mexico, Jun. 2024, pp. 338–354, doi: 10.18653/v1/2024.naacl-long.20.

[38] N. F. Liu et al., "Lost in the middle: How language models use long contexts," *Trans. Assoc. Comput. Linguistics*, vol. 12, pp. 157–173, 2024, doi: 10.1162/tacl_a_00638