

Rancang Bangun Logic.AI: Aplikasi Web *Chatbot* Berbasis *Retrieval-Augmented Generation* untuk Mata Kuliah Logika Informatika

Nasywa Shafa Salsabila^{*1}, Nur Aziezah²

Program Studi Teknologi Rekayasa Perangkat Lunak, Sekolah Vokasi, IPB University
Email: ¹nasywashafa@apps.ipb.ac.id, ²nuraziezah@apps.ipb.ac.id

^{*}Nasywa Shafa Salsabila

Abstrak

Logika Informatika merupakan mata kuliah dengan konsep dan notasi simbolik yang menuntut penalaran berlapis, sehingga mahasiswa kerap mengalami kesulitan ketika belajar mandiri tanpa wadah untuk bertanya. Penelitian ini merancang dan membangun *Logic.AI*, sebuah aplikasi web *chatbot* berbasis *Retrieval-Augmented Generation* (RAG) untuk mata kuliah Logika Informatika, menggunakan metode *Rapid Application Development* (RAD) dengan *Flask*, *LangChain*, *OpenAI GPT-4o mini*, *ChromaDB*, dan *PostgreSQL*. Pengujian fungsionalitas melalui *Black Box Testing* terhadap 56 kasus uji menghasilkan tingkat kelulusan 100%, sedangkan evaluasi kualitas respons menggunakan *BERTScore* terhadap 30 pasang pertanyaan-jawaban menghasilkan rata-rata *F1-Score* 0,8348, melampaui ambang batas 0,80. Hasil tersebut menunjukkan bahwa *Logic.AI* berpotensi menjadi media pendukung pembelajaran mandiri pada mata kuliah Logika Informatika.

Kata kunci: *chatbot* pembelajaran, *large language model*, logika informatika, *rapid application development*, *retrieval augmented generation*.

Abstract

Informatics Logic is a course covering symbolic notation and concepts requiring layered reasoning, which often makes it difficult for students to study independently without a channel to ask questions. This study designs and builds *Logic.AI*, a *Retrieval-Augmented Generation* (RAG)-based web *chatbot* application for the *Informatics Logic* course, using the *Rapid Application Development* (RAD) method with *Flask*, *LangChain*, *OpenAI GPT-4o mini*, *ChromaDB*, and *PostgreSQL*. Functional testing through *Black Box Testing* on 56 test cases yielded a pass rate of 100%, while response quality evaluation using *BERTScore* on 30 question-answer pairs produced an average *F1-Score* of 0.8348, exceeding the 0.80 threshold. These results suggest that *Logic.AI* has potential as a supporting tool for independent study in the *Informatics Logic* course.

Keywords: *large language model*, *learning chatbot*, *informatics logic*, *rapid application development*, *retrieval augmented generation*.

I. PENDAHULUAN

Perkembangan teknologi digital mempercepat transformasi di sektor pendidikan tinggi. Pemanfaatan *Learning Management System* (LMS) dan modul ajar digital kini menjadi standar dalam penyampaian materi perkuliahan [1]. Namun, penerapan materi ajar digital tersebut umumnya masih terbatas pada penyajian informasi bersifat statis dan satu arah [2].

Pada mata kuliah Logika Informatika, mahasiswa dituntut memahami konsep dan notasi simbolik yang menuntut penalaran berlapis [3]. Tidak adanya wadah untuk berinteraksi dan bertanya secara langsung saat belajar mandiri menjadi kendala yang menghambat pemahaman mahasiswa [4]. Ketika menemukan kesulitan pada satu konsep, proses belajar dapat terhenti hingga sesi perkuliahan berikutnya sehingga mengurangi efisiensi pembelajaran [5].

Salah satu pendekatan yang dapat menjembatani kesenjangan ini adalah *chatbot* berbasis *Large Language Model* (LLM). Namun, *chatbot* yang hanya mengandalkan pengetahuan bawaan dari model AI cenderung memberikan respons yang terlalu umum atau tidak sesuai konteks materi kuliah spesifik [6]. Arsitektur *Retrieval-Augmented Generation* (RAG) menjadi alternatif yang lebih sesuai karena memungkinkan pembaruan basis pengetahuan tanpa pelatihan ulang, menghasilkan jawaban yang dapat ditelusuri ke dokumen sumber, serta menurunkan risiko *halusinasi* [7].

Survei global terhadap 23.218 mahasiswa dari 109 negara menunjukkan bahwa mahasiswa pendidikan tinggi telah mengadopsi *ChatGPT* secara luas untuk kegiatan akademik seperti meringkas teks, mencari referensi, dan memahami materi yang kompleks [8]. Seiring kemajuan teknologi LLM,

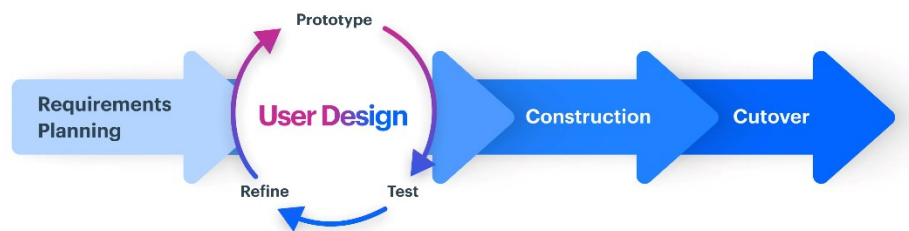
kemampuan *chatbot* pun berkembang pesat dengan arsitektur RAG yang memadukan kemampuan LLM dan sistem *temu kembali* dokumen untuk menghasilkan jawaban berdasarkan basis pengetahuan tertentu [9].

Penelitian ini bertujuan untuk membangun aplikasi *web* pembelajaran berbasis *chatbot* LLM dengan arsitektur RAG untuk mata kuliah Logika Informatika menggunakan metode *Rapid Application Development* (RAD); menguji fungsionalitas aplikasi menggunakan *Black Box Testing* dengan pendekatan *Use Case Based Testing*; dan mengevaluasi kualitas respons *chatbot* menggunakan *BERTScore* dengan metrik *Precision*, *Recall*, dan *F1-Score*.

Studi literatur terhadap lima penelitian sebelumnya menunjukkan *gap* penelitian, yaitu belum ditemukan penelitian yang mengimplementasikan *chatbot* RAG khusus untuk pembelajaran mata kuliah teknis seperti Logika Informatika di pendidikan tinggi Indonesia dengan evaluasi terintegrasi, fungsionalitas (*Black Box Testing*) sekaligus kualitas semantik (*BERTScore*). Penelitian ini hadir untuk mengisi *gap* tersebut.

II. METODE PENELITIAN

Metode pengembangan yang digunakan adalah *Rapid Application Development* (RAD), yang terdiri dari empat fase utama berikut.



Gambar 1. RAD Model Sumber: [10] J. Martin, *Rapid Application Development*. New York: Macmillan Publishing Company, 1991.

2.1. Requirements Planning

Pada fase ini dilakukan wawancara dengan dosen pengampu mata kuliah Logika Informatika untuk mengidentifikasi kebutuhan fungsional dan non-fungsional sistem. Wawancara mencakup aspek pola interaksi mahasiswa, prioritas fitur, prinsip desain *chatbot*, dan kriteria kualitas jawaban. Selain itu dilakukan studi literatur untuk melengkapi kebutuhan teknis sistem. Berdasarkan hasil wawancara, ditetapkan 18 kebutuhan fungsional yang mencakup autentikasi, manajemen materi, manajemen mahasiswa, *chatbot* RAG, riwayat percakapan, dan pemantauan statistik.

2.2. User Design

Fase ini berfokus pada perancangan sistem secara menyeluruh, meliputi: (1) arsitektur sistem RAG dengan alur *indexing*, *retrieval*, dan *generation*; (2) antarmuka pengguna menggunakan UML berupa *use case diagram* dan *activity diagram*; (3) skema basis data PostgreSQL dengan empat entitas utama (*users*, *pdf_materials*, *user_material_progress*, *chat_history*); serta (4) dataset evaluasi *BERTScore* berupa 30 pasang pertanyaan-jawaban yang divalidasi dosen pengampu.

2.3. Construction

Sistem dibangun menggunakan *stack* teknologi berikut: Python sebagai bahasa pemrograman utama, Flask sebagai *micro web framework backend*, LangChain sebagai RAG *framework*, OpenAI GPT-4o mini sebagai LLM melalui *API gateway* OpenRouter, model *text-embedding-3-small* untuk *embedding*, ChromaDB sebagai *vector database*, PostgreSQL sebagai *relational database*, serta HTML/Tailwind CSS/JavaScript untuk antarmuka pengguna.

Implementasi RAG *pipeline* menggunakan LangChain dengan tahapan: (a) *Document Loading* menggunakan *PyPDFLoader*; (b) *Text Splitting* menggunakan *RecursiveCharacterTextSplitter* dengan ukuran 1.000 karakter dan *overlap* 250 karakter; (c) *Embedding* setiap *chunk* menjadi vektor 1.536 dimensi dan disimpan ke ChromaDB; (d) *Retrieval* mengambil *top-k=3 chunk* paling relevan berdasarkan *cosine similarity*; serta (e) *Response Streaming* menggunakan *Server-Sent Events* (SSE).

2.4. Cutover

Pengujian dilakukan pada dua aspek. Pertama, *Black Box Testing* dengan pendekatan *Use Case Based Testing* digunakan untuk menguji fungsionalitas sistem [11]. Setiap skenario uji diturunkan dari *use case* yang telah dirancang, dengan kriteria keberhasilan *pass rate* minimal 90% [12]. Kedua, evaluasi *BERTScore* menggunakan model *xlm-roberta-base* pada Python untuk mengukur kesamaan semantik antara jawaban *chatbot* dan *ground truth* yang divalidasi dosen pengampu. Target nilai rata-rata *F1-Score* adalah di atas 0,80 [13].

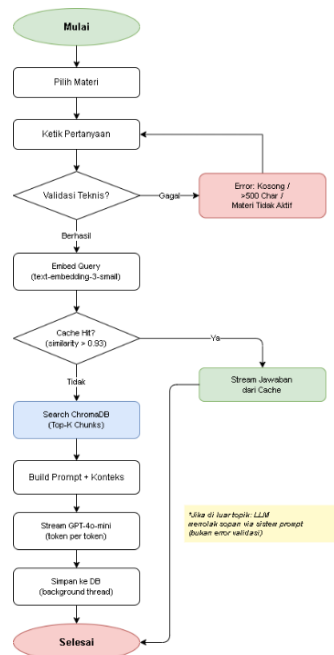
III. HASIL DAN PEMBAHASAN

3.1. Requirements Planning

Wawancara dengan dosen pengampu mata kuliah Logika Informatika menghasilkan temuan bahwa mahasiswa cenderung enggan bertanya saat mengalami kesulitan akibat tekanan sosial, sehingga sistem yang dapat diakses secara mandiri dan anonim menjadi kebutuhan utama. Dosen menekankan pentingnya jawaban yang terstruktur, kemampuan menyertakan contoh konkret dalam setiap respons, serta mekanisme pemberian petunjuk untuk pertanyaan berbentuk soal latihan. Selain itu, sistem harus menggunakan bahasa Indonesia formal namun tidak kaku, dan seluruh respons harus berlandaskan materi Logika Informatika yang telah diunggah oleh dosen. Dari proses ini, berhasil diidentifikasi 18 kebutuhan fungsional sistem untuk dua peran pengguna: Admin dan Mahasiswa.

3.2. User Design

Perancangan sistem mencakup tiga proses bisnis utama: proses Admin (pengelolaan materi PDF dan *embedding* ke ChromaDB), proses Mahasiswa (akses materi dan interaksi *chatbot*), serta proses *chatbot* RAG yang menjadi alur inti sistem. Alur proses *chatbot* RAG dapat dilihat pada Gambar 2.

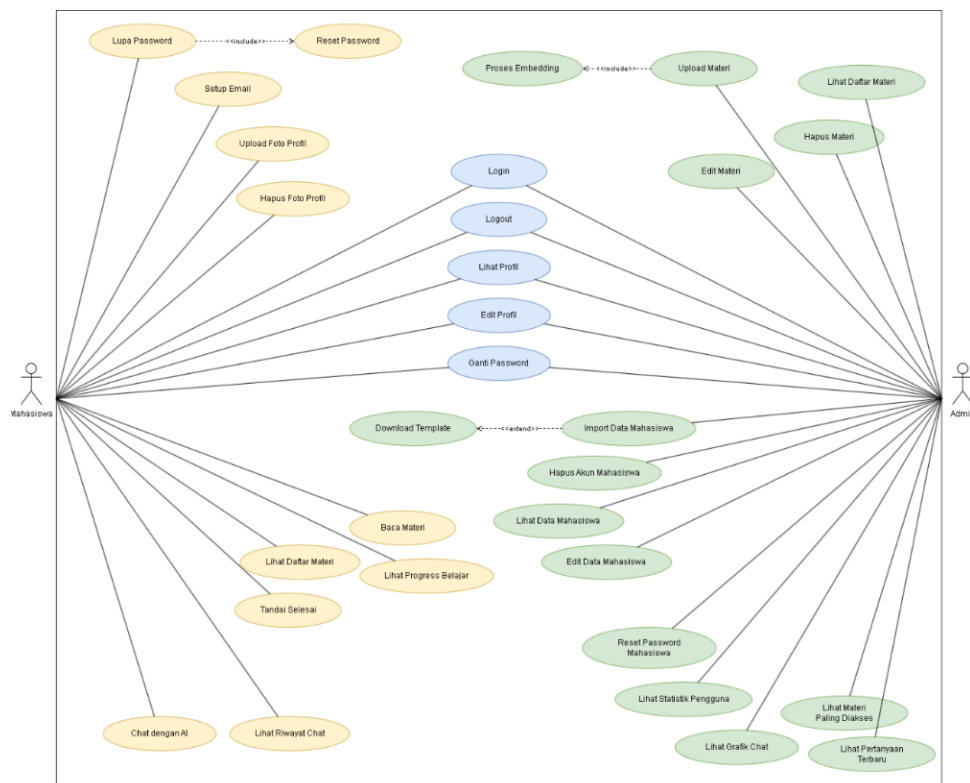


Gambar 2 Proses Bisnis *Chatbot* RAG

Saat pengguna mengirim pertanyaan, sistem melakukan validasi teknis (kosong, >500 karakter, atau materi tidak aktif). Pertanyaan yang valid diubah menjadi vektor menggunakan model *text-embedding-3-small*. Sistem kemudian memeriksa *cache* jawaban dengan ambang batas *similarity* di atas 0,93; jika *cache hit*, jawaban di-*streaming* langsung dari *cache*. Jika *cache miss*, dilakukan *similarity search* ke ChromaDB untuk mengambil top-3 *chunks* paling relevan. Prompt akhir yang menggabungkan pertanyaan, konteks *chunks*, dan *system prompt* kemudian dikirim ke GPT-4o mini untuk menghasilkan respons secara *streaming* token per token. Pertanyaan di luar topik materi ditolak secara sopan oleh LLM melalui *system prompt*.

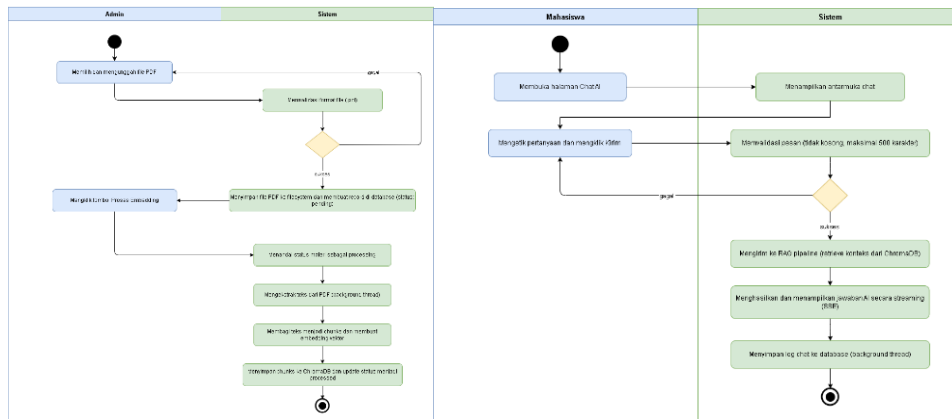
Implementasi RAG *pipeline* menggunakan konfigurasi *chunking* dengan *RecursiveCharacterTextSplitter* berukuran 1.000 karakter dan *overlap* 250 karakter (rasio 25%), model *embedding text-embedding-3-small* untuk menghasilkan vektor 1.536 dimensi, serta *retrieval top-k=3 chunk* berdasarkan *cosine similarity* dari ChromaDB.

Sistem Logic.AI memiliki dua aktor utama dengan total 31 *use case*. Admin memiliki akses eksklusif terhadap manajemen materi, manajemen mahasiswa, dan pemantauan statistik. Mahasiswa memiliki akses eksklusif terhadap fitur pembelajaran dan interaksi chatbot. Diagram *use case* sistem secara keseluruhan dapat dilihat pada Gambar 3.



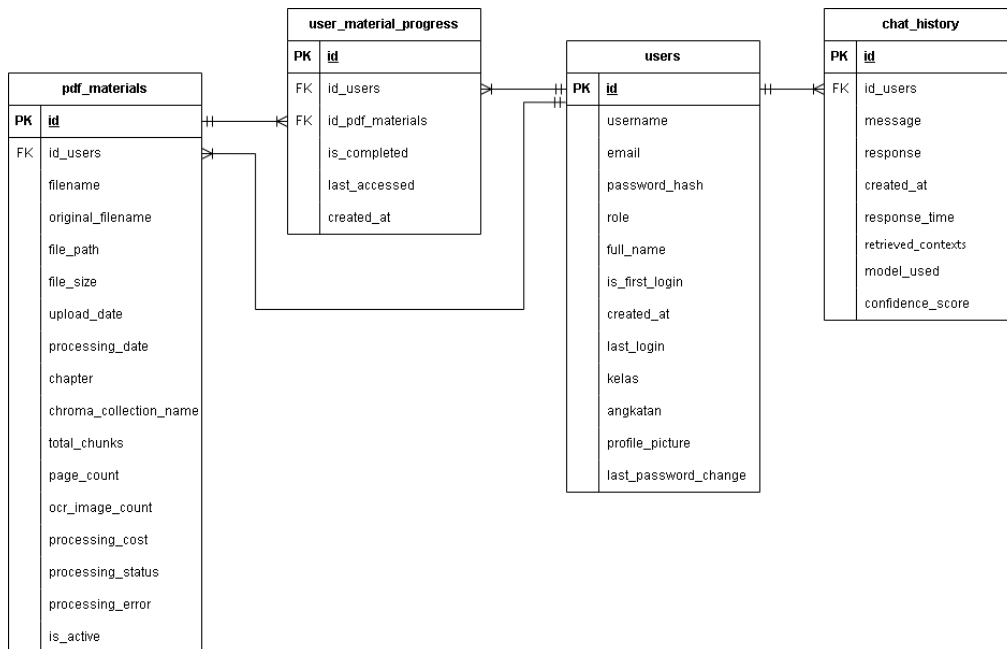
Gambar 3. Use Case Diagram

Alur aktivitas kedua aktor pada fitur utama sistem divisualisasikan melalui *activity diagram*. Gambar 4 menampilkan dua *activity diagram* secara berdampingan: (kiri) alur Unggah dan Proses PDF Materi oleh Admin yang mencakup tahap *extract*, *chunk*, dan *embed* ke ChromaDB; (kanan) alur Chat dengan AI oleh Mahasiswa yang menggambarkan seluruh tahap RAG *pipeline* mulai dari validasi input hingga penyimpanan riwayat percakapan ke PostgreSQL.



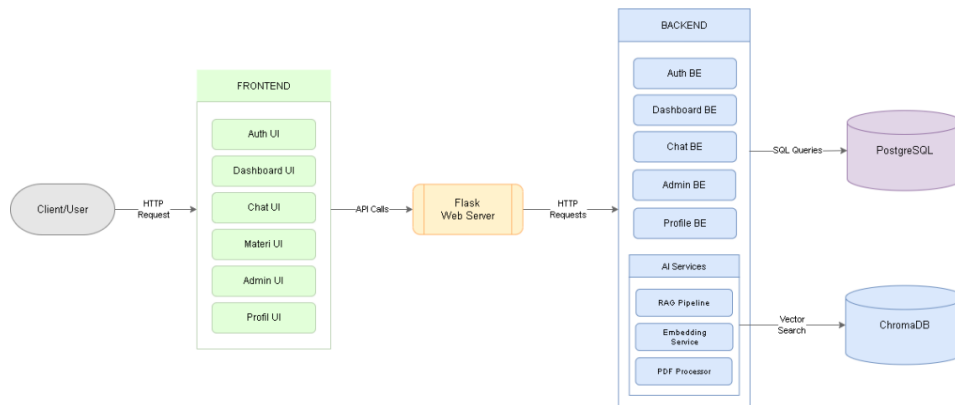
Gambar 4. Activity Diagram (kiri: Admin Unggah & Proses PDF; kanan: Mahasiswa Chat dengan AI)

Basis data sistem dirancang menggunakan PostgreSQL dengan empat entitas utama: *users* (data pengguna Admin dan Mahasiswa), *pdf_materials* (metadata PDF beserta informasi pemrosesan RAG seperti *chroma_collection_name*, *total_chunks*, dan *processing_status*), *user_material_progress* (relasi antara pengguna dan materi dengan atribut *is_completed*), serta *chat_history* (riwayat percakapan dengan atribut *response_time*, *retrieved_contexts*, dan *confidence_score*). Diagram ERD sistem dapat dilihat pada Gambar 5.



Gambar 5. Entity Relationship Diagram (ERD) Sistem Logic.AI

Arsitektur sistem Logic.AI menggunakan pendekatan *client-server* dengan pemisahan yang jelas antara lapisan *frontend* (HTML, Tailwind CSS, JavaScript), *backend* (Flask), dan layanan AI (RAG Pipeline, Embedding Service, PDF Processor) yang terhubung ke ChromaDB melalui *vector search* dan ke PostgreSQL melalui *SQL Queries*. Arsitektur sistem secara keseluruhan dapat dilihat pada Gambar 6.



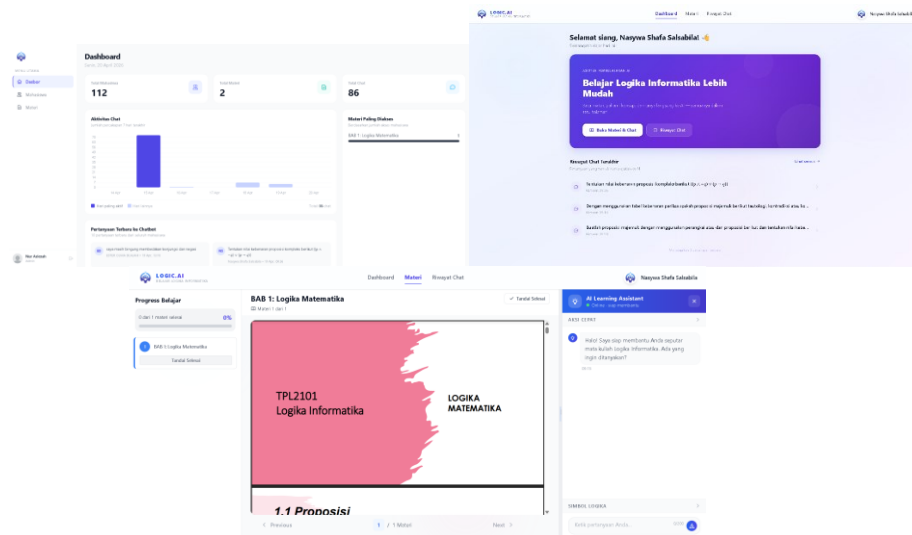
Gambar 6. Arsitektur Sistem Logic.AI

3.3. Construction

Sistem Logic.AI dibangun menggunakan Python dan Flask sebagai *backend*, LangChain untuk RAG *pipeline*, OpenAI GPT-4o mini sebagai LLM via OpenRouter, ChromaDB sebagai *vector database*, dan PostgreSQL sebagai basis data relasional. Antarmuka pengguna dibangun menggunakan HTML, Tailwind CSS, dan JavaScript dengan pendekatan *utility-first* yang menghasilkan tampilan responsif dan konsisten.

Pemilihan setiap komponen teknologi didasarkan pada pertimbangan teknis yang spesifik. *Flask* dipilih karena sifatnya yang *lightweight* dan minimalis yang memberikan fleksibilitas tinggi dalam merancang arsitektur tanpa memaksakan struktur proyek seperti *framework full-stack*, sehingga pengembangan dapat fokus pada logika bisnis utama yaitu implementasi RAG *pipeline*. *LangChain* dipilih karena modularitasnya yang tinggi dalam menyederhanakan kompleksitas RAG *pipeline* melalui komponen modular (*document loaders*, *text splitters*, *retrievers*) yang memungkinkan *swapping* komponen secara fleksibel sesuai karakteristik metode RAD yang iteratif. *OpenAI GPT-4o mini* dipilih sebagai LLM karena menawarkan efektivitas biaya yang optimal untuk skala penelitian tugas akhir, memiliki *context window* hingga 128.000 token untuk memproses banyak *retrieved chunks* tanpa terpotong, serta memberikan dukungan yang baik untuk Bahasa Indonesia. *ChromaDB* dipilih sebagai *vector database* karena arsitektur *local-first* dan fitur *built-in persistence*-nya sangat cocok untuk fase pengembangan dan pengujian iteratif, menjaga data *embedding* tidak hilang saat sistem diuji coba berulang kali. *PostgreSQL* dipilih sebagai RDBMS karena dukungan tipe data yang lebih kaya seperti JSONB yang relevan untuk metadata sistem chatbot, keandalan pada lingkungan *multi-user*, serta dukungan ekosistem Python yang matang. Sementara itu, model *xlm-roberta-base* dipilih untuk evaluasi BERTScore karena kemampuannya sebagai model *multilingual* berbasis Transformer yang mendukung Bahasa Indonesia, sehingga evaluasi kesamaan semantik dapat dilakukan secara akurat pada teks jawaban chatbot yang menggunakan Bahasa Indonesia.

Sistem terdiri dari dua bagian utama. Panel Admin menyediakan *dashboard* statistik (total mahasiswa, total materi, grafik aktivitas chat 7 hari terakhir, 10 pertanyaan terbaru), halaman kelola materi PDF dengan fitur unggah dan *embedding* otomatis ke ChromaDB, serta halaman kelola mahasiswa dengan fitur impor data massal melalui Excel. Portal Mahasiswa menyediakan *dashboard* dengan sapaan personal, halaman materi yang menampilkan PDF *viewer* beserta panel chatbot *floating* yang dapat dibuka-tutup secara independen, *keyboard* simbol logika ($\wedge$, $\vee$, $\neg$, $\rightarrow$, $\leftrightarrow$, $\forall$, $\exists$) untuk memudahkan penulisan notasi simbolik, serta halaman riwayat percakapan. Tampilan antarmuka sistem dapat dilihat pada Gambar 7.



Gambar 7. Antarmuka Sistem Logic.AI: (kiri) *Dashboard Admin*, (kanan) *Dashboard Mahasiswa*,
(bawah) *Halaman Materi & Chatbot*

3.4. Cutover

a) *Black Box Testing*

Black Box Testing dilakukan untuk memverifikasi fungsionalitas sistem tanpa memeriksa struktur internal kode, menggunakan pendekatan *Use Case Based Testing* di mana setiap skenario uji diturunkan langsung dari *use case* yang telah dirancang. Total terdapat 56 kasus uji yang mencakup 31 *use case*, meliputi mekanisme autentikasi, operasi CRUD materi dan mahasiswa, alur RAG chatbot, serta penanganan *error* dan *edge cases*. Rekapitulasi hasil pengujian disajikan pada Tabel 1.

Tabel 1. Rekapitulasi Hasil *Black Box Testing*

Keterangan	Jumlah
Total Kasus Uji	56
Berhasil (<i>Pass</i>)	56
Tidak Berhasil (<i>Fail</i>)	0
Persentase Keberhasilan	100%

Hasil pengujian menunjukkan tingkat keberhasilan 100% dengan 56 kasus uji berhasil (*Pass*). Tingkat keberhasilan 100% melampaui ambang batas minimal 90% yang ditetapkan, menunjukkan bahwa seluruh fungsi utama sistem berjalan sesuai dengan spesifikasi kebutuhan fungsional.

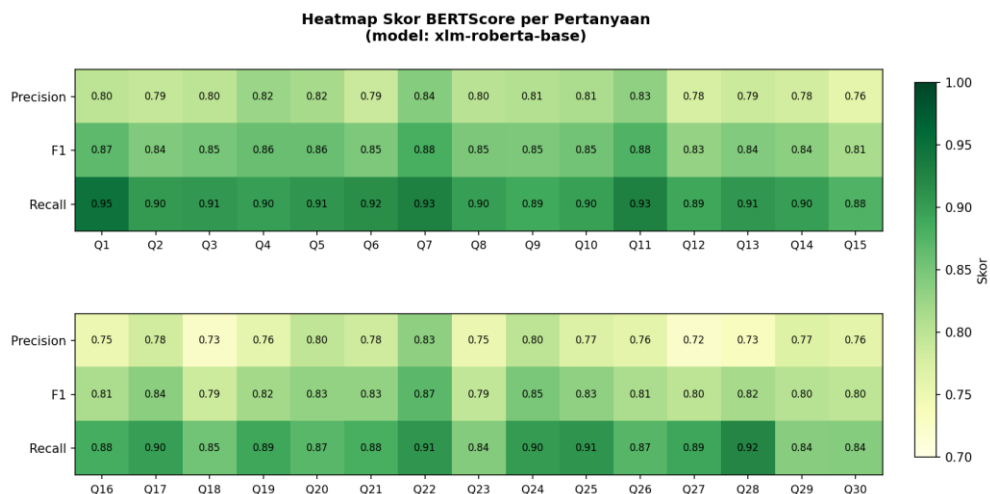
b) *Evaluasi BERTScore*

Evaluasi kualitas respons chatbot dilakukan menggunakan metode *BERTScore* untuk mengukur kesamaan semantik antara jawaban yang dihasilkan chatbot dan jawaban referensi (*ground truth*) yang telah divalidasi oleh dosen pengampu. Evaluasi menggunakan *library bert-score* pada Python dengan model dasar *xlm-roberta-base* untuk mendukung evaluasi teks berbahasa Indonesia. Hasil evaluasi per topik disajikan pada Tabel 2.

Tabel 2 Hasil Evaluasi *BERTScore* per Topik

No	Topik	N	<i>Precision</i>	<i>Recall</i>	F1
1	Proposisi	5	0,8063	0,9130	0,8561
2	Notasi & Nilai Kebenaran	2	0,8134	0,9224	0,8643
3	Negasi	4	0,8134	0,9031	0,8558
4	Konjungsi	4	0,7757	0,8934	0,8304
5	Disjungsi	3	0,7534	0,8791	0,8114
6	Implikasi	6	0,7871	0,8825	0,8320
7	Biimplikasi	3	0,7511	0,8911	0,8149
8	Kesetaraan Logik	3	0,7540	0,8675	0,8059
Rata-rata		30	0,7837	0,8935	0,8348

Distribusi skor *BERTScore* secara individual untuk seluruh 30 pertanyaan (Q1–Q30) divisualisasikan pada Gambar 8.



Gambar 8. Heatmap Skor *BERTScore* per Pertanyaan

Nilai *F1-Score* rata-rata keseluruhan sistem mencapai 0,8348, melampaui ambang batas 0,80 yang ditetapkan. Nilai *Recall* yang tinggi (0,8935) menunjukkan bahwa chatbot berhasil mencakup sebagian besar informasi penting dari jawaban referensi. Nilai *Precision* yang lebih rendah (0,7837) mengindikasikan bahwa chatbot cenderung menghasilkan jawaban yang lebih elaboratif dibandingkan jawaban referensi, karena chatbot menyertakan contoh konkret dan penjelasan bertahap sesuai arahan *system prompt*. Hal ini merupakan karakteristik yang wajar dan justru sesuai dengan tujuan pedagogis sistem.

Topik dengan nilai *F1-Score* tertinggi adalah Notasi & Nilai Kebenaran (0,8643) dan Proposisi (0,8561), yang merupakan topik-topik fundamental dengan konsep yang lebih terdefinisi. Topik dengan nilai *F1-Score* terendah adalah Kesetaraan Logik (0,8059) dan Biimplikasi (0,8149), yang melibatkan transformasi logis lebih kompleks sehingga variasi redaksi jawaban lebih tinggi. Meskipun demikian, seluruh delapan topik mencapai nilai *F1-Score* di atas ambang batas 0,80, yang menunjukkan konsistensi kualitas respons *chatbot* di seluruh cakupan materi Logika Informatika.

Dibandingkan dengan penelitian terdahulu, sistem Logik.AI menghasilkan nilai *F1-Score* sebesar 0,8348, lebih tinggi dibandingkan *chatbot* layanan perpustakaan dengan *F1-Score* 0,7124 [14] maupun *chatbot* tafsir berbasis RAG dengan *F1-Score* 0,7480 [15]. Penelitian lain yang mengembangkan RAG *chatbot* dengan pendekatan *Design Thinking* mencapai 100% pada *Black Box Testing*, namun tidak melengkapi evaluasi kualitas semantik [16]. Penelitian ini menjadi yang pertama mengintegrasikan

evaluasi fungsionalitas dan kualitas semantik pada chatbot RAG untuk mata kuliah teknis di pendidikan tinggi Indonesia.

IV. KESIMPULAN

Aplikasi web pembelajaran berbasis *chatbot* RAG dengan nama Logic.AI telah berhasil dibangun menggunakan metode RAD dengan Flask, LangChain, OpenAI GPT-4o mini, ChromaDB, dan PostgreSQL, serta mengimplementasikan 18 fitur fungsional untuk dua peran pengguna (Admin dan Mahasiswa). Pengujian *Black Box Testing* terhadap 56 kasus uji menghasilkan tingkat keberhasilan 100%. Evaluasi *BERTScore* terhadap 30 pasang pertanyaan-jawaban menghasilkan rata-rata *F1-Score* 0,8348, melampaui ambang batas 0,80, dengan seluruh delapan topik materi memperoleh *F1-Score* di atas 0,80 yang mengindikasikan konsistensi kualitas respons *chatbot* di seluruh cakupan materi Logika Informatika.

Berdasarkan temuan penelitian, disarankan agar: (1) strategi *retrieval* ditingkatkan dengan mengeksplorasi nilai *top-k* yang lebih optimal atau teknik *reranking*, khususnya untuk topik Kesetaraan Logik dan Biimplikasi; (2) pengujian usabilitas menggunakan *System Usability Scale* (SUS) dilakukan dengan melibatkan mahasiswa secara langsung; dan (3) basis pengetahuan sistem diperluas untuk mendukung mata kuliah lain agar Logic.AI dapat berkembang menjadi platform pembelajaran yang lebih komprehensif.

REFERENSI

- [1] C. I. S. Budhayanti, "PENGUNAAN LEARNING MANAGEMENT SYSTEM (LMS) MOODLE PADA PERKULIAHAN PENDIDIKAN MATEMATIKA DI PROGRAM STUDI PENDIDIKAN GURU SEKOLAH DASAR," *Psiko Edukasi*, vol. 21, no. 1, hlm. 50–62, Mei 2023, doi: 10.25170/psikoedukasi.v21i1.4439.
- [2] H. D. Sari, R. Riandi, dan H. K. Surtikanti, "Bahan Ajar Digital Bermuatan Potensi Lokal Untuk Meningkatkan Pemahaman Konsep dan Motivasi Belajar Pada Materi Bioteknologi Konvensional," *J. Basicedu*, vol. 8, no. 1, hlm. 263–276, Jan 2024, doi: 10.31004/basicedu.v8i1.6503.
- [3] L. Mathulhusna, L. Efriyanti, S. Supriadi, dan R. Okra, "Pengaruh Pembelajaran Berbasis Proyek Terhadap Hasil Belajar Logika Algoritma di IAIN Bukittinggi," *Indones. Res. J. Educ.*, vol. 2, no. 3, hlm. 1173–1179, Jul 2022, doi: 10.31004/irje.v2i3.163.
- [4] N. Jannah dan A. Damni, "Investigating Interest and Necessity: A Phenomenological Analysis of Generation Z's Involvement in Islamic Religious Education," *J. Sociol. Agama Indones. JSAI*, vol. 7, no. 1, hlm. 84–98, Mar 2026, doi: 10.22373/jsai.v7i1.9649.
- [5] D. Harefa, "STUDENT DIFFICULTIES IN LEARNING MATHEMATICS," *J. Pendidik. Mat.*, vol. 1, no. 2, hlm. 78–86, Okt 2022, doi: 10.57094/afore.v1i2.431.
- [6] T. N. Fitria, N. E. Simbolon, dan A. Afdaleni, "Chatbots as Online Chat Conversation in the Education Sector," *Int. J. Comput. Inf. Syst. IJCIS*, vol. 4, no. 3, hlm. 93–104, Agu 2023, doi: 10.29040/ijcis.v4i3.116.
- [7] M. Arslan, H. Ghanem, S. Munawar, dan C. Cruz, "A Survey on RAG with LLMs," *Procedia Comput. Sci.*, vol. 246, hlm. 3781–3790, 2024, doi: 10.1016/j.procs.2024.09.178.
- [8] D. Ravšelj *dkk.*, "Higher education students' perceptions of ChatGPT: A global study of early reactions," *PLOS ONE*, vol. 20, no. 2, hlm. e0315011, Feb 2025, doi: 10.1371/journal.pone.0315011.
- [9] J. Swacha dan M. Gracel, "Retrieval-Augmented Generation (RAG) Chatbots for Education: A Survey of Applications," *Appl. Sci.*, vol. 15, no. 8, hlm. 4234, Apr 2025, doi: 10.3390/app15084234.
- [10] J. Martin, *Rapid Application Development*. New York: Macmillan Publishing Company, 1991.
- [11] D. Mardiaty dan Y. Saputra, "IMPLEMENTASI SISTEM INFORMASI MANAJEMEN KLINIK MENGGUNAKAN METODE BLACK BOX TESTING," *J. Inform. Dan Tek. Elektro Terap.*, vol. 13, no. 1, Jan 2025, doi: 10.23960/jitet.v13i1.6015.
- [12] Y. D. Cahyono, L. W. Widiyanti, dan S. I. Pramuningsih, "PENGUJIAN BLACK BOX PADA WEBSITE SISTEM INFORMASI PENGOLAHAN DATA TERITORIAL MENGGUNAKAN METODE USE CASE TESTING UNTUK OPERASI MILITER," vol. 9, no. 2, 2025.
- [13] V. N. D. Abram, "Analisis Kemiripan Menggunakan Metode BERT dan Jaccard Pada Proses Bisnis Bawaslu RI," *Indones. J. Comput. Sci.*, vol. 14, no. 3, Jun 2025, doi: 10.33022/ijcs.v14i3.4890.
- [14] M. A. Qaulan dan P. Adytia, "Pengembangan Chatbot Berbasis AI untuk Mendukung Pelayanan Perpustakaan," vol. 12, no. 1, 2025.

-
- [15] M. R. Maulana, N. S. Harahap, O. Okfalisa, dan Y. Yusra, “Implementasi Chatbot Tafsir Al-Qur’an Menggunakan Chainlit dengan Pendekatan Groq: Implementing a Qur’anic Tafsir Chatbot Through Chainlit with a Groq-Based Approach,” *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 5, no. 3, hlm. 920–929, Jun 2025, doi: 10.57152/malcom.v5i3.2082.
- [16] F. I. Falaq dan N. Saputra, “PENGEMBANGAN RETRIEVAL-AUGMENTED GENERATION (RAG) CHATBOT MENGGUNAKAN METODE DESIGN THINKING UNTUK Mendukung PERSONALIZED LEARNING,” *PINTER J. Pendidik. Tek. Inform. Dan Komput.*, vol. 9, no. 2, hlm. 1–10, Des 2025, doi: 10.21009/pinter.9.2.1.