
Implementation of The Gemini Large Language Models in a Website-Based Chatbot for New Student Admissions at STMIK Widya Cipta Dharma

Faidil Anwar ¹⁾, Ita Arfyanti ²⁾, dan Ahmad Fahrijal Pukeng ³⁾

^{1,3}Teknik Informatika, STMIK Widya Cipta Dharma

²Sistem Informasi, STMIK Widya Cipta Dharma

^{1,2,3}Jl. M. Yamin No. 25, Samarinda, 75123

E-mail: 2243040@wicida.ac.id¹⁾, ita@wicida.ac.id²⁾, pukeng@wicida.ac.id³⁾

ABSTRACT

The New Student Admission (PMB) service at STMIK Widya Cipta Dharma requires comprehensive and continuously accessible information delivery for prospective students. However, conventional services face operational working hour limitations and high staff workloads from repetitive inquiries. While Large Language Models (LLMs) offer interactive communication, they present hallucination risks. To address these challenges, this study designs and implements a website-based PMB chatbot using the Gemini LLM integrated with a Retrieval-Augmented Generation (RAG) architecture to guarantee factual accuracy grounded in official campus documents. Using the Rapid Application Development (RAD) methodology, development progressed through Requirements Planning, User Design, Construction, and Cutover phases. The system employs the gemini-3.1-flash-lite model for natural language processing, complemented by the Qdrant Vector Database to handle semantic search and context indexing for the campus knowledge base. System evaluation confirmed successful implementation across multiple testing phases. Black Box Testing verified that all functional features perform as expected, while White Box Testing confirmed an error-free execution flow with valid cyclomatic complexity paths. Additionally, Beta Testing demonstrated high user acceptance, achieving an 88.8% satisfaction rate among prospective student respondents. Ultimately, combining the Gemini LLM with RAG architecture proves to be a feasible, highly accurate, and effective solution for delivering 24-hour admissions information without the risk of AI hallucination.

Keywords: Large Language Models, Gemini, Chatbot, Retrieval-Augmented Generation, Website

Implementasi Large Language Models Gemini pada Chatbot Penerimaan Mahasiswa Baru STMIK Widya Cipta Dharma Berbasis Website

ABSTRAK

Layanan Penerimaan Mahasiswa Baru (PMB) di STMIK Widya Cipta Dharma memerlukan penyampaian informasi yang komprehensif dan dapat diakses kapan saja oleh calon mahasiswa. Namun, pelayanan konvensional terkendala batas jam kerja operasional serta tingginya beban kerja staf untuk menjawab pertanyaan berulang. Meskipun Large Language Models (LLMs) mampu memberikan interaksi komunikatif, teknologi ini memiliki risiko halusinasi informasi. Untuk mengatasi permasalahan tersebut, penelitian ini bertujuan merancang dan mengimplementasikan chatbot PMB berbasis website menggunakan Gemini LLM yang diintegrasikan dengan arsitektur Retrieval-Augmented Generation (RAG) agar informasi yang disampaikan selalu akurat sesuai dokumen resmi kampus. Pengembangan sistem menerapkan metode Rapid Application Development (RAD) yang meliputi tahapan Requirements Planning, User Design, Construction, dan Cutover. Chatbot ini mengandalkan model gemini-3.1-flash-lite sebagai mesin pemrosesan bahasa alami utama, yang didukung oleh Qdrant Vector Database untuk pencarian semantik dan indeks konteks dokumen basis pengetahuan kampus. Hasil evaluasi menunjukkan bahwa sistem berhasil diimplementasikan secara optimal. Pengujian Black Box memvalidasi keberhasilan seluruh fitur fungsional masukan dan keluaran. Sementara itu, White Box Testing membuktikan alur eksekusi pesan bebas dari kesalahan dengan jalur kompleksitas siklomatis yang valid. Selain itu, evaluasi Beta Testing menghasilkan tingkat kepuasan pengguna sebesar 88,8% dari responden calon mahasiswa. Kesimpulannya, implementasi Gemini LLM dan RAG pada chatbot PMB ini terbukti layak, akurat, serta efektif dalam memberikan layanan informasi 24 jam tanpa risiko halusinasi.

Kata Kunci: Large Language Models, Gemini, Chatbot, Retrieval-Augmented Generation, Website

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi di era digital telah mendorong institusi pendidikan tinggi untuk terus berinovasi dalam menyempurnakan kualitas layanannya. Transformasi digital ini sejalan dengan visi utama STMIK Widya Cipta Dharma, yaitu mewujudkan institusi yang cerdas, teladan, dan bermartabat dalam melaksanakan tridharma perguruan tinggi untuk meningkatkan daya saing bangsa berbasis imtak dan iptek. Upaya mewujudkan visi Smart Campus atau kampus cerdas ini tidak hanya terbatas pada pemanfaatan teknologi di ruang kelas, melainkan juga mencakup modernisasi dan digitalisasi pada sektor pelayanan administrasi yang bersentuhan langsung dengan masyarakat luas.

Salah satu gerbang interaksi paling krusial antara institusi dan calon mahasiswa baru adalah tahapan Penerimaan Mahasiswa Baru (PMB). Pada fase ini, kampus harus mampu menyajikan informasi yang komprehensif terkait ragam keilmuan yang ditawarkan. Saat ini, STMIK Widya Cipta Dharma memiliki pilihan program studi jenjang sarjana yang dirancang untuk menjawab kebutuhan industri teknologi dan bisnis modern, yaitu Teknik Informatika, Sistem Informasi, dan Bisnis Digital. Ketersediaan ragam program studi ini menuntut adanya sebuah pusat layanan informasi yang tanggap agar setiap detail keunggulan akademis dan prosedur pendaftaran dapat tersampaikan dengan jelas kepada calon pendaftar.

Dalam pelaksanaan layanan PMB, pengelola senantiasa berupaya menjaga kelancaran alur komunikasi untuk memfasilitasi calon mahasiswa yang mencari informasi terkait kampus. Staf PMB sering kali dihadapkan pada rutinitas merespons berbagai pertanyaan umum yang serupa dan berulang, mulai dari jadwal pendaftaran, rincian biaya, hingga persyaratan dokumen. Proses pelayanan yang sangat mengandalkan interaksi manusia ini sering kali menyita waktu produktif staf yang seharusnya dapat dialokasikan untuk penyelesaian proses administratif lain yang lebih kompleks, sekaligus memunculkan tantangan dalam menjaga konsistensi kecepatan tanggapan pada periode pendaftaran.

Di samping itu, layanan informasi konvensional memiliki keterbatasan terkait waktu operasional. Calon mahasiswa sering kali mencari dan mengakses informasi akademik di luar jam kerja efektif staf, sehingga mereka harus menunggu hingga keesokan harinya untuk mendapatkan konfirmasi atas pertanyaan yang diajukan. Keterlambatan dalam perolehan informasi esensial ini berpotensi mengurangi efisiensi proses pendaftaran secara keseluruhan. Oleh karena itu, penerapan chatbot menjadi langkah strategis untuk menjembatani kesenjangan komunikasi tersebut, mengingat kemampuannya dalam menyediakan layanan informasi yang interaktif dan bersiaga selama 24 jam penuh tanpa henti.

Pemanfaatan teknologi Large Language Models (LLM) seperti model Gemini menghadirkan dorongan signifikan dalam pengembangan chatbot, memberikan kecerdasan buatan tersebut kemampuan pemrosesan bahasa alami yang tidak kaku dibandingkan sistem tradisional. Model LLM mampu menangkap maksud pengguna dan merangkai kalimat dengan gaya yang komunikatif. Akan tetapi, penggunaan kecerdasan buatan seperti ini memiliki satu kelemahan utama yang disebut halusinasi. Artinya, program ini terkadang bisa memberikan jawaban yang seolah-olah meyakinkan dan benar, padahal informasi yang disampaiakannya keliru atau tidak sesuai dengan fakta.

Sebagai solusi komprehensif untuk memitigasi risiko tersebut dan mengoptimalkan potensi kecerdasan buatan, diperlukan integrasi dengan metode Retrieval-Augmented Generation (RAG). Arsitektur RAG bekerja dengan cara menelusuri data spesifik dari basis pengetahuan dokumen resmi perguruan tinggi terlebih dahulu, lalu menjadikan konteks tersebut sebagai landasan wajib bagi model AI dalam memformulasikan setiap jawabannya. Melalui pendekatan dan penerapan metode ini, chatbot berbasis web yang dikembangkan dapat berinteraksi secara responsif dan menyajikan informasi akademik yang lebih akurat berdasarkan data resmi kampus. Penciptaan infrastruktur pelayanan cerdas, efisien, dan terotomatisasi ini diharapkan dapat mendukung percepatan realisasi visi STMIK Widya Cipta Dharma sebagai kampus cerdas.

2. RUANG LINGKUP

2.1 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka rumusan masalah adalah Bagaimana mengimplementasikan Large Language Models Gemini pada chatbot penerimaan mahasiswa baru STMIK Widya Cipta Dharma berbasis website.

2.2 Batasan Masalah

Adapun batasan masalah dalam penelitian ini untuk menjaga fokus penelitian ini, beberapa batasan masalah ditetapkan sebagai berikut:

1. Versi language model Gemini yang digunakan untuk penelitian ini adalah gemini-3.1-flash-lite.
2. Embedding model yang digunakan untuk proses penyusunan basis pengetahuan adalah model gemini-embedding-001 yang dirancang untuk menghasilkan representasi vektor dari teks.
3. Chatbot ini hanya akan digunakan untuk menjawab pertanyaan seputar penerimaan mahasiswa baru di STMIK Widya Cipta Dharma.
4. Tidak melakukan re-training atau fine-tuning pada model Gemini, melainkan menggunakan teknik RAG untuk meningkatkan performa model dalam menjawab pertanyaan terkait penerimaan mahasiswa baru di STMIK Widya Cipta Dharma.
5. Chatbot yang dikembangkan hanya menyajikan informasi, tidak melakukan proses pendaftaran atau

interaksi lainnya yang memerlukan akses ke sistem internal STMIK Widya Cipta Dharma.

2.3 Tujuan

Tujuan dari penelitian ini adalah merancang dan mengimplementasikan chatbot berbasis website menggunakan teknologi LLM Gemini pada layanan Penerimaan Mahasiswa Baru di STMIK Widya Cipta Dharma. Sistem percakapan pintar ini dibangun dengan mengintegrasikan arsitektur RAG untuk memastikan setiap informasi akademik yang disampaikan kepada calon mahasiswa selalu akurat, relevan, dan bersumber langsung dari dokumen resmi institusi.

2.4 Manfaat

1. Bagi Calon Mahasiswa Baru

Penelitian ini memberikan manfaat bagi calon mahasiswa baru berupa kemudahan memperoleh informasi PMB secara cepat dan mudah diakses kapan saja, termasuk di luar jam kerja. Melalui chatbot berbasis web, calon pendaftar dapat menanyakan jadwal pendaftaran, biaya, persyaratan dokumen, pilihan program studi, serta alur registrasi dengan bahasa yang natural. Dukungan basis pengetahuan dari dokumen resmi kampus membantu memastikan jawaban yang diberikan tetap selaras dengan informasi yang berlaku sehingga mengurangi risiko kekeliruan. Hal ini dapat meningkatkan kejelasan informasi, menghemat waktu pencarian, dan membantu calon mahasiswa mengambil keputusan yang lebih tepat.

2. Bagi Perguruan Tinggi

Penelitian ini bermanfaat bagi perguruan tinggi dalam meningkatkan kualitas layanan informasi PMB melalui penyediaan layanan komunikasi yang responsif, konsisten, dan tersedia 24 jam. Chatbot dapat mengurangi beban kerja staf PMB dalam menangani pertanyaan yang bersifat berulang sehingga staf dapat fokus pada layanan yang bersifat administratif yang memerlukan penanganan khusus. Dengan rujukan informasi yang didukung oleh sumber resmi, jawaban yang dihasilkan menjadi lebih terkontrol sehingga membantu menjaga kredibilitas informasi institusi. Selain itu, interaksi pengguna dapat menjadi masukan untuk evaluasi kebutuhan informasi, perbaikan materi sosialisasi, serta mendukung percepatan realisasi layanan Smart Campus.

3. BAHAN DAN METODE

3.1 Large Language Models

Menurut Sylvia (2025), Large Language Models (LLM) adalah model kecerdasan buatan berbasis arsitektur transformer yang dilatih menggunakan data teks dalam jumlah yang masif. Pelatihan dengan skala besar ini memungkinkan model untuk dapat memahami, menganalisis, dan menghasilkan bahasa alami secara kontekstual dengan tingkat kecanggihan yang menyerupai interaksi manusia.

Menurut Mu'alifah & Haviana (2025), LLM merupakan inovasi signifikan dalam ranah pemrosesan bahasa alami atau Natural Language Processing (NLP) yang secara khusus difungsikan untuk menjalankan berbagai tugas otomatisasi teks. Dengan memanfaatkan teknologi deep learning, LLM mampu melatih dirinya sendiri dalam mengenali pola dan fitur dari data teks untuk merangkum dokumen, menjawab interaktif, hingga menyusun informasi secara koheren tanpa banyak campur tangan manual.

Sementara itu, menurut Talaohu dkk. (2025), pemanfaatan LLM telah menjadi standar teknologi mutakhir yang sangat efektif dalam pengembangan agen virtual seperti chatbot. Integrasi model LLM ke dalam chatbot memungkinkan sistem untuk memahami input bahasa alami pengguna secara lebih mendalam dan menyajikan informasi balasan secara responsif, interaktif, serta jauh lebih dinamis dibandingkan sistem terdahulu.

Dari beberapa pendapat di atas, dapat disimpulkan bahwa Large Language Models (LLM) adalah sistem kecerdasan buatan tingkat lanjut berbasis jaringan saraf transformer yang dilatih dengan korpus data teks yang sangat besar. Teknologi ini memberikan kapabilitas pada komputer untuk memahami, memproses, dan memproduksi bahasa manusia secara luwes dan kontekstual, sehingga sangat ideal untuk diimplementasikan sebagai perangkat utama pada aplikasi layanan informasi terotomatisasi atau chatbot.

3.2 Gemini

Menurut Gemini Team Google dkk. (2025), Gemini adalah model kecerdasan buatan berarsitektur multimodal tingkat lanjut yang sejak awal dirancang untuk memiliki kemampuan penalaran komprehensif. Sistem ini dilatih secara terpadu untuk memahami teks, gambar, audio, dan video secara bersamaan, serta dirilis dalam beberapa versi, agar dapat disesuaikan dengan berbagai tingkat komputasi, mulai dari tugas yang sangat kompleks hingga pengoperasian langsung pada perangkat (on-device).

Sementara itu, menurut Rachmat & Kesuma (2024), Gemini merupakan inovasi LLM yang sangat adaptif dan andal untuk diimplementasikan dalam pengembangan aplikasi chatbot. Penggunaan teknologi Gemini terbukti mampu secara signifikan meningkatkan kemampuan chatbot dalam memproses bahasa alami, sehingga respons yang diberikan kepada pengguna menjadi lebih akurat, natural, dan interaktif.

Dapat disimpulkan bahwa Gemini adalah teknologi LLM multimodal mutakhir ciptaan Google yang memiliki kemampuan komprehensif dalam memahami, memproses, dan menghasilkan bahasa alami beserta jenis data lainnya. Fleksibilitasnya yang tinggi, variasi model, serta kemampuannya dalam penalaran konteks menjadikan Gemini sebagai instrumen kecerdasan buatan yang sangat tangguh dan ideal untuk dijadikan sebagai otak utama dalam perancangan aplikasi chatbot yang responsif.

3.3 Mahasiswa

Dalam Kamus Besar Bahasa Indonesia (KBBI) mahasiswa didefinisikan sebagai seseorang yang sedang belajar di Perguruan Tinggi (Kamus Besar Bahasa Indonesia, t.t.). Menurut Supriantino (2024), mahasiswa merupakan individu dalam masyarakat yang memegang status akademik tertinggi. Kedudukan eksklusif ini menempatkan mahasiswa bukan lagi sekadar sebagai individu yang belajar secara pasif, melainkan bertransformasi menjadi agen perubahan sekaligus agen pemberdayaan yang berperan krusial dalam pembangunan fisik maupun non-fisik sebuah bangsa. Dari penjelasan tersebut, dapat disimpulkan bahwa mahasiswa adalah kaum inteligensia yang mandiri, kritis, dan berwawasan luas.

3.4 Chatbot

Menurut Talaohu dkk. (2025), chatbot adalah teknologi berbasis kecerdasan buatan yang secara khusus dirancang untuk merespons dan menyajikan informasi kepada pengguna secara interaktif dan otomatis. Sistem ini memungkinkan pengguna untuk mendapatkan jawaban instan layaknya berkomunikasi langsung dengan petugas manusia.

Sementara itu, menurut Anwar dkk. (2025), chatbot adalah inovasi teknologi percakapan yang bertindak sebagai asisten virtual cerdas dalam memfasilitasi kebutuhan interaksi digital. Penggunaan chatbot berfungsi secara strategis untuk mengatasi keterbatasan layanan tenaga manusia secara konvensional, yakni dengan menyediakan akses informasi yang selalu siaga selama 24 jam penuh, mengotomatisasi penanganan pertanyaan yang bersifat repetitif, serta mempercepat waktu tanggap pelayanan.

Dapat disimpulkan bahwa chatbot adalah program komputer atau agen virtual berbasis kecerdasan buatan yang dirancang untuk mensimulasikan percakapan interaktif dua arah dengan manusia melalui teks. Teknologi ini berfungsi sebagai asisten layanan cerdas yang mampu memahami instruksi bahasa alami pengguna dan meresponsnya secara instan tanpa batasan waktu operasional.

3.5 Website

Menurut Rifai dkk. (2025), website adalah kumpulan halaman yang menyimpan informasi dalam bentuk data digital, termasuk teks, gambar, video, audio, dan animasi, yang dapat diakses melalui internet. Platform ini berfungsi sebagai media penyampaian informasi yang dinamis dan terintegrasi dalam suatu sistem jaringan.

Sementara itu, Menurut Maharani dkk. (2024), website merupakan suatu kumpulan halaman yang saling terkait melalui tautan, di mana setiap halamannya mampu menampilkan serta memuat beragam informasi digital. Format informasi tersebut sangat bervariasi, mencakup teks, gambar, video, audio, animasi, hingga kombinasi interaktif dari keseluruhan elemen tersebut yang terintegrasi di dalam sebuah domain.

Dapat disimpulkan bahwa website adalah sekumpulan halaman digital berbasis multimedia yang

saling terhubung di dalam suatu domain dan dapat diakses secara global melalui internet menggunakan peramban web. Dalam implementasinya, website tidak sekadar berfungsi sebagai wadah penyimpanan data statis, melainkan juga sebagai media penyampaian informasi yang dinamis, interaktif, dan mampu membangun kredibilitas serta kepercayaan publik terhadap institusi atau organisasi yang mengelolanya.

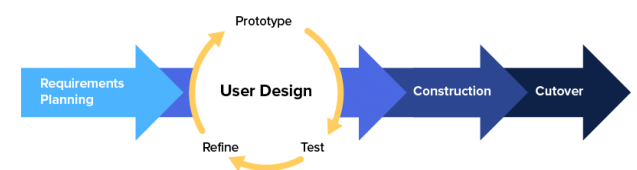
3.6 Retrieval Augmented Generation

Menurut Fanani (2025), Retrieval-Augmented Generation (RAG) adalah pendekatan inovatif dalam teknologi kecerdasan buatan yang secara sinergis menggabungkan kekuatan Large Language Model (LLM) dengan kemampuan sistem untuk melakukan pencarian informasi secara eksternal. Sistem RAG ini beroperasi dengan cara menelusuri data atau informasi yang relevan dari sebuah basis pengetahuan eksternal terlebih dahulu sebelum merumuskan respons, sehingga jawaban yang dikembalikan kepada pengguna menjadi jauh lebih akurat dan terikat pada konteks yang benar.

Sementara itu, menurut Pujiono dkk. (2024), metode RAG bekerja melalui perpaduan antara retrieval dengan generation. Melalui tahapan ekstraksi dokumen berbasis regulasi atau pangkalan data ini, model kecerdasan buatan dapat memahami konteks pertanyaan secara lebih utuh dan memanfaatkan ruang lingkup informasi yang lebih luas guna memberikan balasan yang jauh lebih spesifik dan tepat sasaran.

Dapat disimpulkan bahwa Retrieval-Augmented Generation (RAG) adalah suatu kerangka kerja kecerdasan buatan yang mengkombinasikan modul pencari informasi (retriever) dengan modul pembuat teks (generator). Pendekatan ini berfungsi sebagai penyaring sekaligus penyuplai konteks dari pangkalan data eksternal ke dalam mesin bahasa, memastikan setiap kalimat yang dihasilkan sepenuhnya bersandar pada fakta yang transparan dan meminimalkan halusinasi pada interaksi digital.

3.7 Metode Rapid Application Development



Gambar 1. Model Rapid Application Development

Figure 1. Rapid Application Development Model

Menurut Wicaksono & Suryawan (2023), Rapid Application Development (RAD) adalah sebuah pendekatan berorientasi objek dalam pengembangan sistem yang mencakup metode pengembangan serta perangkat lunak pendukung lainnya. Metode ini menekankan pada siklus pengembangan yang sangat pendek dan singkat untuk menghasilkan sistem yang fungsional. Tujuan utama dari metode RAD adalah mempersingkat waktu yang biasanya dibutuhkan dalam

siklus hidup pengembangan sistem tradisional antara pembuatan spesifikasi dan pengiriman sistem akhir kepada pengguna. Ada beberapa tahapan utama dalam metode RAD, yaitu:

1. Requirements Planning. Pada tahap ini, pengguna dan analis bertemu untuk mengidentifikasi tujuan-tujuan dari aplikasi atau sistem serta mengidentifikasi syarat-syarat informasi yang muncul dari tujuan-tujuan tersebut. Proses ini tidak hanya berfokus pada pengumpulan kebutuhan fungsional, tetapi juga mencakup identifikasi batasan sistem, kendala teknis, serta prioritas fitur yang akan dikembangkan agar keselarasan antara ekspektasi pengguna dan kapabilitas teknis dapat ditetapkan sejak awal.
2. User Design. Selama tahap ini, pengguna berinteraksi intensif dengan analis sistem untuk mengembangkan model-model dan prototipe yang mewakili semua proses sistem, input, dan output. Desain ini biasanya dilakukan melalui sesi-sesi interaktif yang melibatkan pengguna secara langsung, seperti workshop, sehingga rancangan antarmuka dan alur kerja sistem mencerminkan kebutuhan nyata pengguna sebelum implementasi penuh dilakukan.
3. Construction. Tahap ini melibatkan pembangunan sistem secara iteratif dengan menggunakan teknik pemrograman yang cepat dan efisien. Prototipe yang dihasilkan selama tahap desain digunakan sebagai dasar untuk pengembangan lebih lanjut. Pada tahap ini pengguna tetap berpartisipasi aktif dan masih dapat memberikan usulan perubahan saat prototipe sedang dikembangkan, memastikan bahwa setiap siklus iterasi menghasilkan komponen sistem yang semakin mendekati kebutuhan akhir yang sesungguhnya.
4. Cutover. Tahap akhir ini mencakup pengujian sistem secara menyeluruh, migrasi dan konversi data dari sistem lama ke sistem baru, serta penyelenggaraan pelatihan bagi para pengguna akhir. Setelah seluruh proses validasi dan serah terima selesai dilakukan, sistem akhirnya siap untuk dioperasikan secara penuh dalam lingkungan produksi yang sesungguhnya.

3.8 Blackbox Testing

Menurut Kartono dkk. (2024), Black Box Testing merupakan suatu metode evaluasi perangkat lunak yang berfokus sepenuhnya pada spesifikasi fungsionalitas sistem tanpa perlu meninjau atau mengetahui struktur kode internal di belakangnya. Pengujian ini bertujuan untuk memastikan bahwa setiap fitur masukan (input) dan keluaran (output) pada aplikasi berjalan sesuai dengan ekspektasi desain dan kebutuhan operasional pengguna akhir.

3.9 Whitebox Testing

Menurut Suryanta dkk. (2025), White Box Testing adalah suatu metode evaluasi perangkat lunak yang berfokus pada peninjauan detail prosedural dan struktur logika internal dari baris kode program. Pengujian ini mengharuskan penguji untuk memiliki pemahaman

mendalam tentang algoritma sistem, sehingga setiap jalur eksekusi, percabangan logika, dan perulangan di dalam source code dapat dipastikan berjalan sesuai dengan rancangan awal tanpa adanya bug tersembunyi. metode pengujian White Box sangat krusial untuk mengukur kompleksitas kode serta menentukan semua kemungkinan jalur independen yang dapat dilalui oleh program saat dieksekusi. Melalui pemetaan logika sistem ke dalam bentuk Flow Graph, pengembang dapat memvalidasi setiap batasan kondisi untuk meminimalisasi risiko kegagalan sistem.

Dalam penerapannya, jumlah skenario pengujian yang harus dilakukan pada teknik Basis Path ditentukan dengan menghitung nilai Cyclomatic Complexity (Kompleksitas Siklomatis). Rumus matematika yang digunakan untuk menghitung nilai kompleksitas ini berdasarkan grafik alir adalah sebagai berikut:

$$V(G) = E - N + 2$$

1. $V(G)$ = Nilai Cyclomatic Complexity, jumlah jalur independen.
2. E = Jumlah total edges, garis atau panah penghubung antar-titik pada Flow Graph.
3. N = Jumlah total nodes, titik atau simpul logika pada Flow Graph.

3.10 Beta Testing

Menurut Hakim dkk. (2024), Beta Testing merupakan tahapan evaluasi perangkat lunak yang dilakukan langsung oleh calon pengguna akhir (end-user) di luar lingkungan pengembang. Pengujian ini bertujuan untuk mengukur tingkat kualitas, kinerja, serta keandalan aplikasi saat dioperasikan secara nyata, sekaligus menjaring umpan balik terkait pengalaman pengguna sebelum perangkat lunak tersebut dirilis secara resmi ke publik.

Metode Beta Testing memegang peranan penting dalam tahapan validasi penerimaan pengguna (User Acceptance) terhadap sebuah aplikasi. Melalui penyebaran kuesioner atau pengumpulan tanggapan pasca-penggunaan, pengembang dapat mengetahui seberapa baik antarmuka dan fitur aplikasi yang telah dibangun mampu memenuhi kebutuhan serta ekspektasi fungsionalitas dari perspektif audiens sasaran.

Dalam pelaksanaannya, hasil kuesioner dari fase pengujian ini sering kali diukur secara kuantitatif seperti menggunakan Skala Likert untuk mengonversi persepsi pengguna menjadi data numerik yang objektif. Untuk mengetahui tingkat kelayakan atau penerimaan sistem secara keseluruhan, hasil tanggapan responden dihitung menggunakan rumus persentase persetujuan sebagai berikut:

$$P = \frac{\sum x}{\sum x_{max}} \times 100\%$$

1. P = Persentase tingkat kelayakan atau penerimaan sistem berdasarkan tanggapan pengguna.

2. x = Jumlah total skor yang diberikan oleh responden untuk setiap item pertanyaan dalam kuesioner.
3. $x_{\max}$ = Jumlah total skor ideal atau maksimum.

4. PEMBAHASAN

Pada bab ini akan dibahas mengenai tahapan-tahapan yang dilakukan dalam penelitian ini, mulai dari perencanaan hingga pengujian. Berikut adalah tahapan-tahapan yang dilakukan dalam penelitian ini:

4.1 Perencanaan (Requirements Planning)

4.1.2 Analisis Kebutuhan Sistem

1. Kebutuhan Fungsional

Berdasarkan analisis yang telah dilakukan, kebutuhan fungsional dari sistem chatbot ini adalah sebagai berikut:

- a. Calon mahasiswa baru (user) dapat memberikan pertanyaan kepada chatbot.
- b. Chatbot dapat menjawab pertanyaan dari user.
- c. Admin dapat mengelola basis pengetahuan (knowledge base).
- d. Admin dapat melihat riwayat percakapan user dengan chatbot.

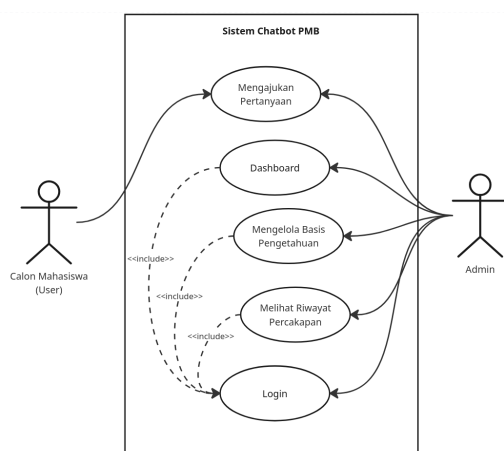
2. Kebutuhan Non-Fungsional

Adapun kebutuhan non-fungsional dari sistem chatbot ini adalah sebagai berikut:

- a. Sistem dikembangkan menggunakan laptop dengan sistem operasi Linux.
- b. Visual Studio Code digunakan sebagai teks editor.
- c. TypeScript akan dikompilasi dan dijalankan dengan runtime Bun.
- d. Hono digunakan sebagai backend framework.
- e. ReactJS dan TailwindCSS digunakan sebagai frontend framework.
- f. PostgreSQL digunakan sebagai basis data relasional.
- g. Qdrant digunakan sebagai basis data vektor.
- h. Gemini API digunakan sebagai Large Language Model (LLM).
- i. Firefox Browser digunakan sebagai web browser untuk melakukan pengujian.

4.2 Pemodelan (User Design)

4.2.1 Use Case Diagram



Gambar 2. Diagram Use Case
Figure 2. Use Case Diagram

Aktor:

1. User: Calon mahasiswa baru yang menggunakan chatbot untuk mendapatkan informasi terkait PMB STMIK Widya Cipta Dharma.
2. Admin: Pengelola sistem chatbot yang bertanggung jawab untuk mengelola basis pengetahuan dan memantau riwayat percakapan user dengan chatbot.

Use Case untuk User:

1. Mengajukan Pertanyaan – Calon mahasiswa baru (user) dapat memasukkan teks pertanyaan seputar informasi PMB dan menerima jawaban langsung dari chatbot.

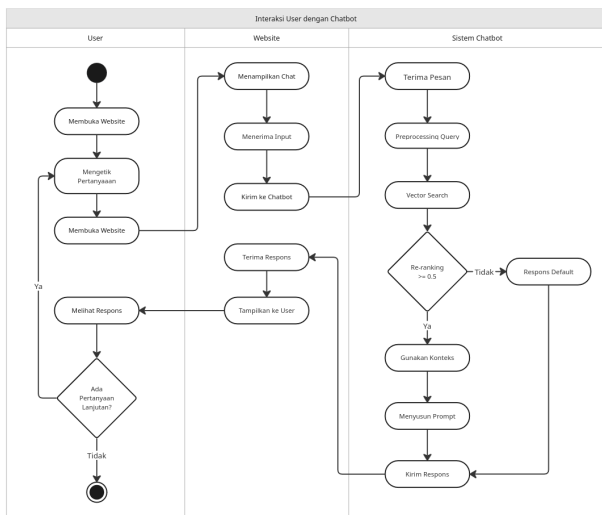
Use Case untuk Admin:

1. Mengajukan Pertanyaan – Admin dapat mengajukan pertanyaan kepada chatbot untuk menguji respons sistem.
2. Login – Admin wajib masuk ke dalam sistem menggunakan akun terdaftar sebelum dapat mengakses fitur pengelolaan chatbot.
3. Dashboard – Menampilkan halaman utama yang berisi ringkasan informasi sistem setelah admin berhasil melakukan login.
4. Mengelola Basis Pengetahuan – Admin dapat menambah, mengubah, atau mengelola dokumen dan data informasi PMB yang akan dijadikan acuan rujukan pengetahuan oleh chatbot.
5. Melihat Riwayat Percakapan – Admin dapat melihat, memantau, dan meninjau histori pesan obrolan antara user dan chatbot.

Pada Gambar 2, terdapat relasi `<<include>>` yang menghubungkan use case Dashboard, Mengelola Basis Pengetahuan, dan Melihat Riwayat Percakapan menuju use case Login. Relasi ini menunjukkan bahwa proses login bersifat wajib (mandatory). Artinya, Admin diharuskan untuk berhasil melakukan autentikasi atau login terlebih dahulu sebelum sistem mengizinkan Admin untuk mengakses dashboard, memodifikasi basis pengetahuan, maupun memantau riwayat percakapan.

4.2.2 Activity Diagram

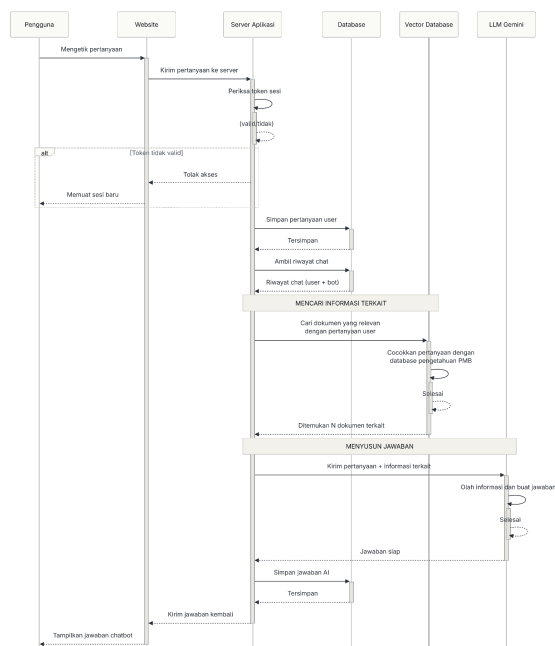
Gambar 3 menampilkan alur kerja *Activity Diagram* interaksi antara calon mahasiswa dan *chatbot* PMB. Proses dimulai saat pengguna mengirimkan pertanyaan akademik melalui antarmuka *website*. Sistem kemudian menerima pesan, melakukan *query preprocessing*, dan menjalankan *vector search* pada *Vector Database*. Melalui tahap *re-ranking* dengan ambang batas skor kemiripan 0,5, sistem akan menggunakan respons bawaan (*default*) jika skor tidak terpenuhi, atau menyusun *prompt* berkonteks untuk dikirim ke LLM jika skor sesuai. Setelah *chatbot* menampilkan jawaban akhir di *website*, pengguna dapat memilih untuk mengajukan pertanyaan lanjutan atau mengakhiri sesi.



Gambar 3. Activity Diagram Interaksi dengan Chatbot

Figure 3. Interaction with Chatbot Activity Diagram

4.2.3 Sequence Diagram



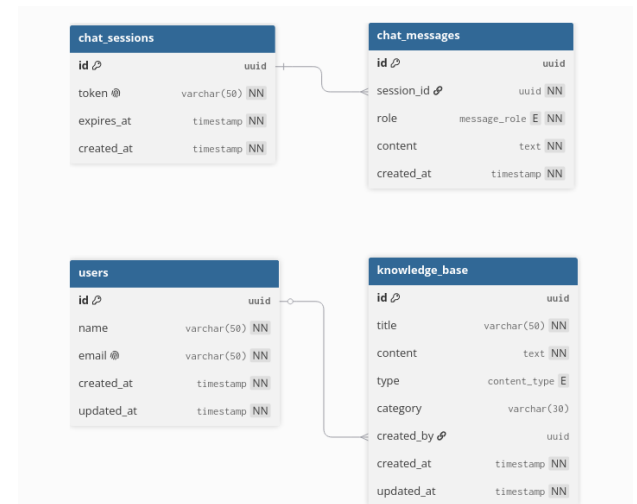
Gambar 4. Sequence Diagram Percakapan User dengan Chatbot

Figure 4. User Chat with Chatbot Sequence Diagram

Gambar 4 menggambarkan pemodelan interaksi antara calon mahasiswa dan chatbot berbasis RAG. Proses dimulai saat user mengirim pertanyaan melalui website. Server menyimpan query ke database, mengambil riwayat percakapan, lalu mencocokkan pertanyaan ke Vector Database untuk mengekstraksi dokumen PMB yang relevan. Selanjutnya, server menyusun prompt berisi pertanyaan dan konteks dokumen, lalu mengirimkannya ke LLM Gemini.

Jawaban dari Gemini diteruskan server ke website untuk ditampilkan. Mekanisme ini memastikan jawaban yang akurat sesuai pedoman kampus sekaligus mendukung percakapan multi-turn.

4.2.4 Entity Relationship Diagram



Gambar 5. Entity Relationship Diagram

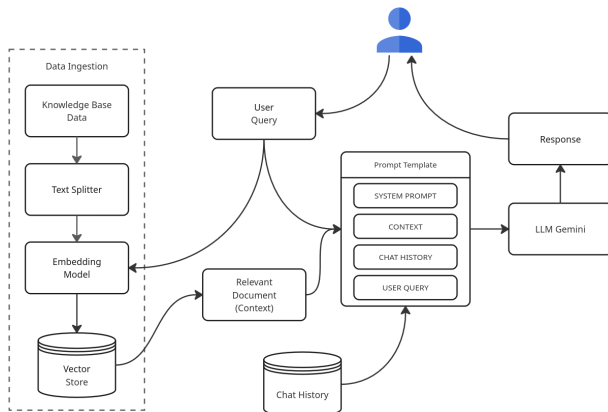
Figure 5. Entity Relationship Diagram

Berdasarkan Gambar 5, dapat dilihat ERD yang digunakan untuk memodelkan struktur penyimpanan data relasional serta memvisualisasikan keterkaitan antar entitas di dalam basis data sistem chatbot. Arsitektur basis data relasional sistem ini dibangun oleh empat entitas utama, yaitu tabel users, knowledge_base, chat_sessions, dan chat_messages. Dalam operasionalnya, terdapat relasi one-to-many antara entitas chat_sessions dan chat_messages, yang berarti satu sesi percakapan dapat memiliki dan menampung banyak pesan obrolan di dalamnya. Selain itu, terdapat pula relasi one-to-many antara entitas users dan knowledge_base, yang menunjukkan bahwa satu entitas user (admin) dapat membuat, menambah, serta mengelola banyak data basis pengetahuan yang nantinya akan digunakan sebagai rujukan jawaban oleh chatbot.

4.3 Implementasi (Construction)

4.3.1 Implementasi Arsitektur RAG

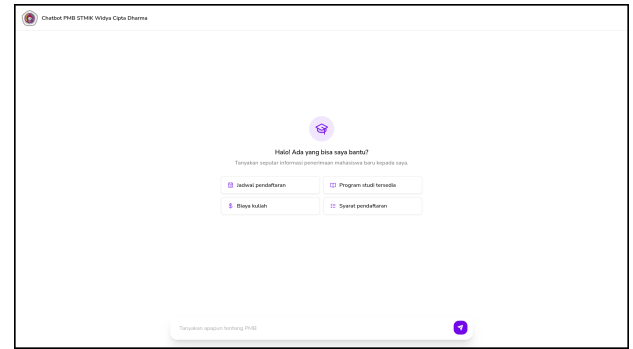
Tahap implementasi arsitektur Retrieval-Augmented Generation (RAG) merupakan bagian inti yang mengatur bagaimana sistem kecerdasan buatan membaca, mencari, dan memproses informasi untuk menghasilkan jawaban. Alur sistem ini bermula dari fase pemasukan data (data ingestion), di mana dokumen resmi kampus yang berformat PDF diproses dan dipotong menjadi bagian-bagian teks yang lebih ringkas menggunakan modul Text Splitter. Potongan-potongan teks tersebut kemudian diubah menjadi representasi numerik (vector) menggunakan Embedding Model dan disimpan ke dalam Vector Store sebagai basis pengetahuan utama chatbot.



Gambar 5. Arsitektur RAG
Figure 5. RAG Architecture

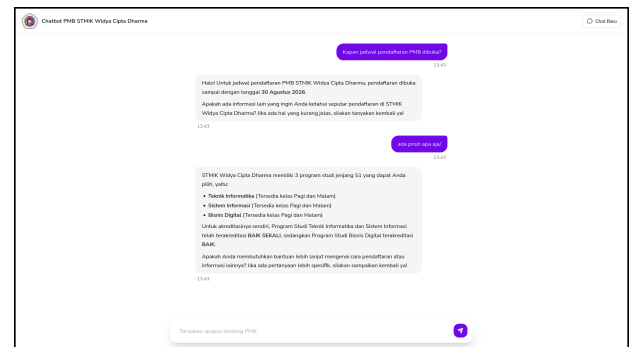
Pada Gambar 5, ditampilkan arsitektur Retrieval-Augmented Generation (RAG) yang beroperasi saat pengguna berinteraksi dengan sistem. Ketika pengguna mengirimkan pertanyaan (User Query), sistem akan meneruskan pertanyaan tersebut ke Embedding Model untuk mengubah pertanyaan menjadi representasi vektor. Vektor pertanyaan ini kemudian digunakan untuk melakukan pencarian (search) di dalam Vector Store guna menemukan dokumen yang paling relevan dengan pertanyaan pengguna. Setelah dokumen relevan ditemukan, sistem akan mengekstrak konten dokumen tersebut sebagai konteks rujukan. Selanjutnya, sistem menyusun Prompt Template yang berisi instruksi dasar (System Prompt), konteks dokumen, riwayat percakapan sebelumnya (Chat History) yang diambil dari database, dan pertanyaan asli dari pengguna. Rangkaian instruksi ini kemudian dikirimkan ke LLM Gemini untuk diproses menjadi jawaban akhir (Response), yang kemudian ditampilkan kembali ke layar pengguna. Penerapan mekanisme penggabungan instruksi ini memiliki peran yang sangat esensial dalam menjaga keandalan sistem. Dengan disisipkannya Relevant Document ke dalam Prompt Template, model kecerdasan buatan dipandu untuk bersandar sepenuhnya pada pedoman resmi kampus, sehingga risiko terjadinya halusinasi atau penyampaian informasi yang keliru dapat diminimalisasi secara signifikan. Selain itu, integrasi Chat History memungkinkan chatbot untuk memahami alur percakapan secara berkesinambungan (multi-turn conversation), sehingga pengguna dapat mengajukan pertanyaan lanjutan tanpa kehilangan konteks pembicaraan sebelumnya. Melalui keseluruhan proses arsitektural ini, jawaban akhir yang diproduksi oleh LLM Gemini tidak hanya terjamin keakuratan faktanya, tetapi juga disajikan dengan gaya bahasa yang natural, komunikatif, dan interaktif bagi calon mahasiswa baru..

4.3.2 Halaman Chat



Gambar 6. Halaman Chat (Utama)
Figure 6. Chat Page (Main)

Pada Gambar 6, ditampilkan antarmuka utama halaman chat yang akan dilihat pertama kali oleh calon mahasiswa baru ketika mengakses layanan. Halaman ini menyajikan teks sapaan awal yang ramah untuk memberikan kesan pertama yang baik kepada pengguna. Selain itu, terdapat pula beberapa pintasan pertanyaan umum yang dapat langsung diklik oleh pengguna untuk mempercepat proses pencarian informasi. Di bagian bawah halaman, tersedia kolom teks yang cukup luas bagi calon mahasiswa untuk mengetikkan pertanyaan secara mandiri. Antarmuka ini dirancang agar bersih dan fokus sehingga pengguna dapat langsung berinteraksi dengan mudah. Tampilan yang sederhana ini bertujuan untuk meningkatkan kenyamanan dan kemudahan akses informasi bagi siapa pun yang baru pertama kali membuka laman website penerimaan mahasiswa baru tersebut.

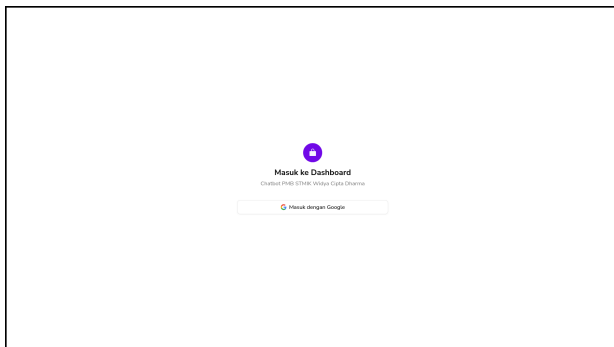


Gambar 7. Halaman Chat (Sesi Percakapan)
Figure 7. Chat Page (Chat Session)

Kemudian pada Gambar 7, ditampilkan sesi percakapan yang sedang berlangsung antara pengguna dan sistem chatbot. Pada halaman ini, terlihat sistem mampu memberikan tanggapan informasi secara luwes, natural, dan terstruktur sesuai dengan data resmi kampus. Sebagai contoh, sistem dapat menyajikan daftar program studi ke dalam bentuk rincian poin-poin yang rapi agar mudah dibaca oleh calon mahasiswa. Jawaban yang dihasilkan oleh chatbot tidak hanya sekadar memberikan teks, tetapi juga menyertakan saran pertanyaan lanjutan

untuk membantu calon mahasiswa mengeksplorasi informasi lebih mendalam.

4.3.3 Halaman Login Dashboard

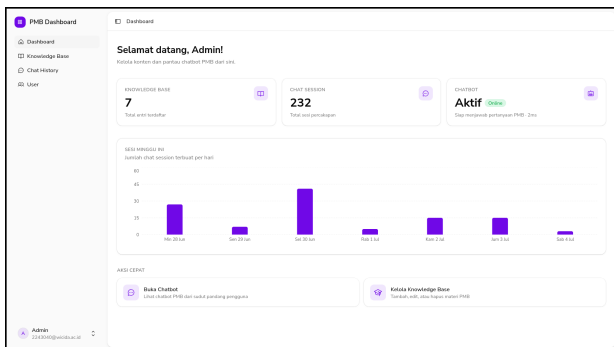


Gambar 8. Halaman Login Dashboard

Figure 8. Dashboard Login Page

Pada Gambar 8, ditampilkan halaman khusus yang berfungsi sebagai gerbang masuk bagi pengelola atau admin untuk mengakses sistem kendali aplikasi. Halaman ini didesain dengan tingkat keamanan yang memadai melalui integrasi tombol autentikasi Google, sehingga sistem dapat memverifikasi identitas pengguna secara cepat dan akurat. Penggunaan metode Login dengan Google ini memberikan kemudahan bagi admin karena tidak perlu lagi mengingat kata sandi tambahan yang terpisah, cukup menggunakan akun yang sudah dimiliki. Setelah proses verifikasi identitas di sisi server berhasil dijalankan, admin akan langsung dialihkan secara otomatis ke bagian utama kendali.

4.3.4 Halaman Dashboard



Gambar 9. Halaman Dashboard

Figure 9. Dashboard Page

Pada Gambar 9, ditampilkan halaman dashboard utama yang menjadi pusat kendali setelah admin berhasil melewati proses autentikasi. Halaman ini menyajikan ringkasan kinerja sistem secara visual dan menyeluruh, sehingga admin dapat memantau kondisi aplikasi hanya dengan sekali liat. Informasi yang ditampilkan mencakup data statistik jumlah sesi percakapan yang terjadi, akumulasi dokumen pengetahuan yang sudah tersimpan, serta grafik aktivitas pengguna dalam kurun waktu tertentu. Selain itu, terdapat indikator yang memberikan

informasi secara real-time mengenai status ketersediaan layanan chatbot saat ini. Dashboard ini dirancang secara khusus untuk mempermudah admin dalam melakukan pengambilan keputusan berbasis data, guna memastikan bahwa layanan informasi kepada calon mahasiswa dapat berjalan optimal setiap harinya.

4.3.5 Halaman Pengelolaan Basis Pengetahuan

No	Judul	Tipe	Kategori	Dibuat	Aksi
1	Prosedur PMSB SMIK WIDYADARMAS 2022 Final	T. File	pendaftaran	03 Jul 2020, 03:12	
2	Jadwal Pendaftaran	T. Teks	pendaftaran	28 Jun 2020, 21:20	
3	Akreditasi	T. Teks	akreditasi	28 Jun 2020, 13:05	
4	Kontak atau Nomorhang PMSB Widyadarmas	T. Teks	kontak	28 Jun 2020, 12:45	
5	Biaya Kuliah	T. Teks	biaya kuliah	28 Jun 2020, 12:38	
6	FAQ PMSB Widyadarmas	T. Teks	faq	28 Jun 2020, 12:32	
7	Panduan Mengisi Form Pendaftaran	T. Teks	form pendaftaran	28 Jun 2020, 12:28	

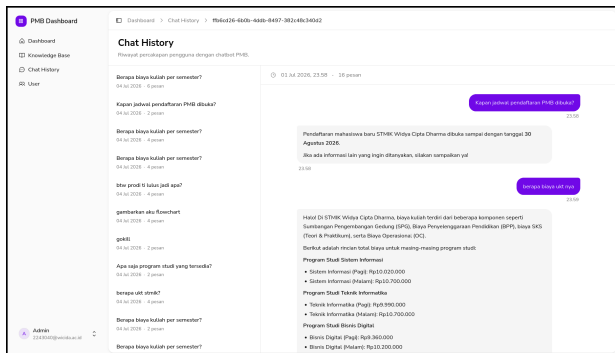
Gambar 10. Halaman Pengelolaan Basis Pengetahuan

Figure 10. Knowledge Base Management Page

Pada Gambar 10, ditampilkan halaman untuk pengelolaan basis pengetahuan yang menjadi “otak” dari sistem chatbot ini. Halaman ini menyediakan fitur bagi admin untuk menambah, mengubah, atau menghapus materi informasi kampus agar jawaban yang diberikan AI senantiasa terkini. Terdapat fitur untuk memasukkan informasi baru secara manual melalui teks, atau mengunggah dokumen referensi dalam bentuk berkas PDF yang akan diekstraksi isinya secara otomatis. Setelah perubahan dilakukan, admin dapat menjalankan tombol sinkronisasi yang memerintahkan sistem untuk memperbarui indeks vektor agar AI dapat langsung memahami materi terbaru tersebut.

4.3.6 Halaman Riwayat Percakapan

Pada Gambar 11, ditampilkan halaman riwayat percakapan yang berfungsi sebagai pusat peninjauan seluruh aktivitas interaksi pengguna. Halaman ini menampilkan daftar histori pesan antara calon mahasiswa dan chatbot secara terurut berdasarkan waktu. Admin dapat membuka setiap sesi percakapan untuk meninjau secara mendalam bagaimana sistem memberikan tanggapan atas pertanyaan-pertanyaan yang masuk. Fitur ini sangat bermanfaat sebagai sarana evaluasi, di mana admin bisa menemukan topik-topik informasi yang sering ditanyakan atau mendeteksi jika terdapat jawaban dari AI yang kurang sesuai. Dengan melakukan pengawasan berkala melalui halaman ini, admin dapat terus meningkatkan kualitas layanan komunikasi institusi agar lebih responsif, tepat sasaran, dan memberikan pengalaman yang lebih baik bagi calon mahasiswa.



Gambar 11. Halaman Riwayat Percakapan
Figure 11. Chat History Page

4.4 Pengujian (Cutover)

4.4.1 Blackbox Testing

Tabel 1. Hasil Pengujian Blackbox
Table 1. Blackbox Testing Results

Aktor	Skenario	Hasil Yang Diharapkan	Hasil Pengujian
User	Menanyakan daftar program studi yang tersedia	Chatbot menampilkan daftar program studi beserta akreditasinya	Berhasil
User	Menanyakan Persyaratan pendaftaran calon mahasiswa baru	Chatbot memberikan jawaban berisi persyaratan pendaftaran yang akurat sesuai dokumen resmi	Berhasil
User	Menanyakan jadwal lengkap kegiatan PMB	Chatbot menampilkan jadwal pelaksanaan kegiatan PMB dengan rincian yang jelas dan terstruktur	Berhasil
User	Menanyakan komponen biaya pendaftaran dan rinciannya	Chatbot memberikan informasi detail tentang biaya yang harus dibayarkan oleh calon mahasiswa baru	Berhasil
User	Menanyakan informasi tentang beasiswa yang tersedia	Chatbot memberikan jawaban mengenai program beasiswa yang ditawarkan	Berhasil
User	Menanyakan pertanyaan di	Chatbot memberikan	Berhasil

luar konteks PMB (misal: cuaca, olahraga, dll)

respons yang sopan menolak atau mengalihkan ke topik PMB

Admin	Login ke sistem menggunakan akun Google yang terdaftar	Admin berhasil masuk ke dashboard	Berhasil
Admin	Menambahkan informasi baru ke basis pengetahuan secara manual	Data berhasil disimpan ke database	Berhasil
Admin	Mengunggah dokumen PDF ke basis pengetahuan	Dokumen berhasil diunggah, teks diekstrak, dan tersimpan di database	Berhasil
Admin	Melakukan sinkronisasi basis pengetahuan ke Vector Database	Data berhasil di sinkronisasi dan dapat diakses oleh chatbot	Berhasil

4.4.2 Whitebox Testing

Pada tahap ini, pengujian dilakukan dengan fokus pada struktur internal dan logika kode program. Berikut adalah analisis program yang diuji pada pengujian whitebox:

```

61 export const sendChat = async (sessionId: string, token: string, input: SendChatInput): Promise<SendChatResult> => {
62   const chatSession = await chatRepository.findSessionById(sessionId)
63   if (!chatSession) {
64     return { status: 'not_found' }
65   }
66   if (!token) {
67     return { status: 'invalid_token' }
68   }
69   const validateToken = await validateToken(token)
70   if (!validateToken || validateToken.sessionId !== sessionId) {
71     return { status: 'invalid_token' }
72   }
73   await chatRepository.createMessage({
74     sessionId,
75     role: 'user',
76     content: input.message
77   })
78   const chatMessages = await chatRepository.findMessagesBySessionId(sessionId)
79   const contents = await chatRepository.findContents(chatMessages)
80   const botResponse = await chatAi(contents)
81   await chatRepository.createMessage({
82     sessionId,
83     role: 'model',
84     content: botResponse
85   })
86   return {
87     status: 'ok',
88     message: botResponse
89   }
90 }

```

Gambar 12. Kode Program Mengirim Pertanyaan ke Chatbot

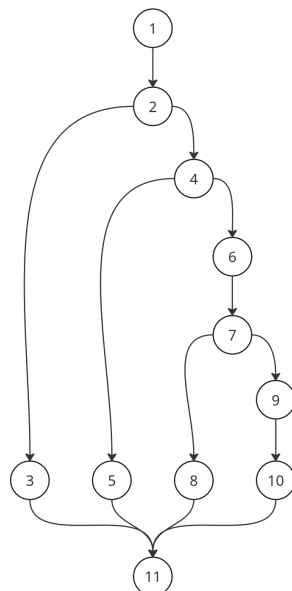
Figure 12. Source Code for Sending Questions to Chatbot

Tabel 2. Deskripsi Kode Program Mengirim Pertanyaan ke Chatbot

Table 2. Description of Source Code Sending Questions to Chatbot

Node	Baris Kode	Description
------	------------	-------------

1	61	Mulai dan memanggil fungsi Chat Repository untuk mencari sesi chat yang sesuai dengan sessionId yang diberikan.
2	64	Validasi apakah sesi chat ditemukan.
3	65	Jika sesi chat tidak ditemukan, mengembalikan status not_found.
4	68	Validasi apakah token diberikan.
5	69	Jika token tidak diberikan, mengembalikan status invalid_token.
6	72	Memanggil fungsi validateToken untuk memeriksa keabsahan token.
7	73	Validasi apakah token valid dan apakah sessionId yang terkait dengan token sesuai dengan sessionId yang diberikan.
8	74	Jika token tidak valid atau sessionId tidak sesuai, mengembalikan status invalid_token.
9	77-91	Proses menyimpan pesan user, mengambil seluruh pesan dalam sesi chat, memetakan pesan-pesan tersebut menjadi format yang sesuai untuk diproses oleh model AI, dan memanggil fungsi chatAI untuk mendapatkan respons dari chatbot, dan menyimpan respons tersebut ke dalam database.
10	93-96	Mengembalikan status 'ok' beserta pesan balasan dari chatbot.
11	97	Selesai



Gambar 13. Flowgraph Mengirim Pertanyaan ke Chatbot

Figure 13. Source Code for Sending Questions to Chatbot

Berdasarkan struktur Flowgraph Gambar 13, diperoleh informasi mengenai jumlah Node (N) sebanyak 11 dan jumlah Edge (E) sebanyak 13.

Perhitungan nilai kompleksitas siklomatis (Cyclomatic Complexity) dilakukan untuk menentukan jumlah jalur uji independen minimal yang harus dieksekusi menggunakan rumus sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 13 - 11 + 2 = 4 \text{ Jalur}$$

Melalui hasil perhitungan kompleksitas tersebut, diperoleh 4 jalur uji independen yang merepresentasikan seluruh kemungkinan alur eksekusi logika di dalam fungsi sendChat. Keempat jalur tersebut adalah sebagai berikut:

1. Node: 1 - 2 - 3 - 11
Keterangan: Kondisi ketika chat sesi tidak ditemukan
2. Node: 1 - 2 - 4 - 5 - 11
Keterangan: Kondisi ketika chat token kosong
3. Node: 1 - 2 - 4 - 6 - 7 - 8 - 11
Keterangan: Kondisi ketika chat token tidak valid
4. Node: 1 - 2 - 4 - 6 - 7 - 9 - 10 - 11
Keterangan: Kondisi ketika chat sukses

Selanjutnya, dilakukan pengujian rancangan test case terhadap masing-masing jalur uji independen tersebut untuk memvalidasi kesesuaian antara output sistem dengan hasil yang diharapkan. Pada Jalur 4, pengujian juga mencakup kemampuan sistem dalam memuat riwayat percakapan sebelumnya agar model AI dapat merespons beberapa pertanyaan secara berkesinambungan. Rincian skenario dan hasil pengujian dapat dilihat pada Tabel 3.

Tabel 3. Test Case Pengujian Fungsi Mengirim Pertanyaan ke Chatbot

Table 3. Test Case the Function of Sending Questions to the Chatbot

Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
Mengirim pertanyaan ke chatbot dengan sessionId yang tidak valid	Sistem mengembalikan status not_found dan tidak memproses pertanyaan lebih lanjut	Berhasil
Mengirim pertanyaan ke chatbot tanpa menyertakan chat token	Sistem mengembalikan status invalid_token dan tidak memproses pertanyaan lebih lanjut	Berhasil
Mengirim pertanyaan ke chatbot dengan token yang tidak valid atau tidak sesuai dengan sessionId	Sistem mengembalikan status invalid_token dan tidak memproses pertanyaan lebih lanjut	Berhasil

Mengirim pertanyaan ke chatbot dengan sessionId dan token yang valid, serta memuat riwayat percakapan sebelumnya jika ada	Sistem menyimpan pesan user, mengambil riwayat percakapan, memproses pertanyaan melalui model AI, menyimpan balasan dari chatbot, dan mengembalikan status 'ok' beserta pesan balasan dari chatbot	Berhasil
---	--	----------

4.4.3 Beta Testing

Pada tahap ini, dilakukan pengujian sistem chatbot oleh calon mahasiswa baru yang menjadi responden. Tujuan dari pengujian ini adalah untuk mengevaluasi sejauh mana sistem dapat memenuhi kebutuhan pengguna dan memberikan pengalaman yang memuaskan. Responden diminta untuk mengisi kuesioner yang berisi pertanyaan-pertanyaan terkait kepuasan mereka terhadap layanan chatbot PMB.

Tabel 4. Skala Penilaian

Table 4. Rating Scale

Nilai	Keterangan
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Netral
4	Setuju
5	Sangat Setuju

Tabel 5. Skala Penilaian

Table 5. Questionnaire Question List

Kode	Kuesioner
P1	Apakah tampilan website chatbot ini menarik dan mudah untuk digunakan?
P2	Apakah navigasi pada website chatbot mudah dipahami oleh Anda sebagai pengguna baru?
P3	Apakah proses mengetik dan mengirim pesan ke chatbot berjalan lancar?
P4	Apakah chatbot mampu memberikan jawaban yang akurat terkait informasi Penerimaan Mahasiswa Baru?
P5	Apakah jawaban yang diberikan oleh chatbot relevan dan sesuai dengan konteks pertanyaan yang Anda ajukan?
P6	Apakah gaya bahasa yang digunakan oleh chatbot terasa natural, sopan, dan mudah dipahami?

P7	Apakah informasi dari chatbot ini sudah cukup untuk menjawab kebingungan Anda sebagai calon pendaftar?
P8	Apakah chatbot membalas pertanyaan Anda dengan waktu yang cepat?
P9	Apakah website chatbot dapat diakses dan berjalan lancar tanpa adanya error?
P10	Secara keseluruhan, apakah Anda merasa terbantu dan puas dengan layanan chatbot PMB ini?

Tabel 6. Hasil Kuesioner Beta Testing

Table 6. Beta Testing Questionnaire Results

Respon den	Pertanyaan										Jumlah
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	
1.	4	4	4	3	4	5	3	4	4	4	39
2.	4	5	5	5	5	5	5	5	5	5	49
3.	5	5	5	4	5	5	4	5	3	5	45
4.	4	4	5	4	3	5	5	4	5	5	44
5.	4	3	5	3	3	4	3	5	4	4	38
6.	4	5	5	5	5	5	5	5	5	5	49
7.	5	5	5	5	5	5	5	5	5	5	50
8.	4	4	4	4	4	4	3	4	3	4	38
9.	5	5	4	5	4	4	4	3	5	5	44
10.	4	5	5	5	5	5	4	5	5	5	48

Jumlah responden yang berpartisipasi dalam Beta Testing ini adalah 10 orang. Berdasarkan hasil kuesioner, diperoleh total skor sebesar 444 dari 500 poin maksimal. Dengan demikian, dapat dihitung persentase kepuasan pengguna dengan menggunakan rumus berikut:

$$P = \frac{\sum x}{\sum x_{max}} \times 100\%$$

$\sum x = 444$ (total skor yang diperoleh dari seluruh pertanyaan kuesioner)

$\sum x_{max} = 500$ (total skor maksimal dari seluruh pertanyaan kuesioner)

$$P = \left(\frac{444}{500} \right) \times 100\% = 88.8\%$$

Dari hasil perhitungan di atas, diperoleh persentase kepuasan pengguna sebesar 88.8%. Hal ini menunjukkan bahwa sistem chatbot PMB telah berhasil memenuhi kebutuhan dan harapan calon mahasiswa baru.

5. KESIMPULAN

Sistem chatbot berbasis website berhasil dibangun menggunakan model Gemini yang dipadukan dengan arsitektur Retrieval-Augmented Generation (RAG) dan basis data vektor Qdrant, sehingga mampu menjawab pertanyaan seputar PMB secara natural, akurat, serta sesuai dengan dokumen resmi kampus. Untuk mendukung pengelolaannya, sistem menyediakan dashboard khusus yang memudahkan admin dalam mengelola basis pengetahuan, baik melalui pengetikan

manual maupun unggah berkas PDF referensi kampus, sekaligus memantau riwayat percakapan pengguna untuk evaluasi layanan. Selain itu, hasil pengujian Black Box dan White Box menunjukkan seluruh fungsionalitas sistem berjalan dengan baik tanpa bug, serta diperkuat oleh pengujian Beta Testing yang memperoleh tingkat kepuasan pengguna sebesar 88,8%, membuktikan bahwa aplikasi ini sangat layak dan bermanfaat bagi calon mahasiswa baru.

6. SARAN

1. Penelitian saat ini terbatas pada chatbot sebagai pusat penyedia informasi PMB saja. Pada penelitian selanjutnya, diharapkan dapat mengembangkan sistem ini menjadi Agentic AI (agen kecerdasan buatan otonom). Dengan Agentic AI, sistem tidak hanya sekedar menjawab pertanyaan, tetapi diberi kapabilitas untuk melakukan tindakan nyata, seperti memandu dan memproses pendaftaran calon mahasiswa baru, memvalidasi kelengkapan berkas yang diunggah, hingga menjadwalkan sesi ujian seleksi secara otomatis ke sistem internal kampus.
2. Saat ini chatbot baru dapat diakses melalui platform website resmi PMB. Disarankan agar sistem selanjutnya dapat dikembangkan secara multi-platform, yaitu dengan mengintegrasikan layanan chatbot ini ke platform perpesanan instan yang sering digunakan oleh calon mahasiswa seperti WhatsApp dan Telegram, guna meningkatkan kemudahan serta jangkauan akses informasi secara 24 jam penuh.
3. Penelitian ini berfokus pada penggunaan model Gemini (gemini-3.1-flash-lite). Untuk pengembangan dan analisis ilmiah lebih lanjut, disarankan untuk melakukan studi perbandingan (benchmark) performa arsitektur RAG menggunakan Large Language Models (LLM) selain Gemini seperti GPT-5, DeepSeek. Perbandingan dapat diukur berdasarkan kecepatan respons dan tingkat akurasi pemahaman konteks (retrieval accuracy).

7. REFERENSI

Anwar, R., Pratiwi, H., & Wahyuni. (2025). Application of Large Language Model for New Student Admission Chatbot. *IJISTECH (International Journal of Information System and Technology)*, 8(6). <https://doi.org/10.30645/ijistech.v8i6.379>

Fanani, I. (2025). IMPLEMENTASI RETRIEVAL AUGMENTED GENERATION UNTUK EVALUASI PROPOSAL TUGAS AKHIR MAHASISWA. *Jurnal Teknologi Komputer dan Informatika*, 3(2). <https://doi.org/10.59820/tekomin.v3i2.336>

Gemini Team Google, Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., Silver, D., Johnson, M., Antonoglou, I., Schrittwieser, J., Glaese, A., Chen, J., Pitler, E., Lillicrap, T., Lazaridou, A., ... Vinyals,

O. (2025, Mei). Gemini: A Family of Highly Capable Multimodal Models. *arXiv*. <https://doi.org/10.48550/arXiv.2312.11805>

Hakim, H. L., Faqih, D., Deva, D., Hudaya, I. F., & Ilyas, M. N. (2024). Pengujian Alpha dan Beta Testing Pada Aplikasi TIJE. *TeknoIS: Jurnal Ilmiah Teknologi Informasi dan Sains*, 14(2), 285-295. <https://doi.org/10.36350/jbs.v14i2.265>

Kamus Besar Bahasa Indonesia. (t.t.). Arti kata mahasiswa - Kamus Besar Bahasa Indonesia (KBBI) Online. <https://kbbi.web.id/mahasiswa>

Kartono, F. K., Nursaadah, S., Nugroho, M. R., Tama, D. A., Mashudi, F. A., Wicaksono, A., & Nasir, M. (2024). Pengujian Black Box Testing Pada Sistem Website Osha Snack: Pendekatan Teknik Boundary Value Analysis. *Jurnal Kridatama Sains dan Teknologi*, 6(2), 754-766. <https://doi.org/10.53863/kst.v6i02.1407>

Maharani, T. P., Permatasari, H., & Purwanto, E. (2024). Rancang Bangun Sistem Pengelolaan Magang Berbasis Website Pada Digimizu Digital Management. *Journal Computer Science and Information Systems: J-Cosys*, 4(2). <https://doi.org/10.53514/jco.v4i2.524>

Mu'alifah, E., & Haviana, S. F. C. (2025). Perancangan Sistem Peringkasan Artikel Ilmiah Menggunakan Grobid Dan Karge Language Models (LLM) Berbasis SciBERT. *Jurnal Rekayasa Sistem Informasi dan Teknologi*, 2(3), 1094-1102. <https://doi.org/10.70248/jrsit.v2i3.1917>

Pujiono, I., Agtyaputra, I. M., & Ruldeviyani, Y. (2024). IMPLEMENTING RETRIEVAL-AUGMENTED GENERATION AND VECTOR DATABASES FOR CHATBOTS IN PUBLIC SERVICES AGENCIES CONTEXT. *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)*, 10(1), 216-223. <https://doi.org/10.33480/jitk.v10i1.5572>

Rachmat, N., & Kesuma, D. P. (2024). Implementasi LLM Gemini Pada Pengembangan Aplikasi Chatbot Berbasis Android. *Jurnal Ilmu Komputer (JUIK)*, 4(1), 40-52. <https://doi.org/10.31314/juik.v4i1.2831>

Rifai, M., Akbar, R., & Nurrisma. (2025). Sistem Pendukung Keputusan Penerimaan Calon Siswa Baru MIN 40 Aceh Besar Berbasis Web. *Jurnal Manajemen Sistem Informasi*, 4(1), 40-56. <https://doi.org/10.59431/jmasif.v4i1.494>

Supriantino, T. (2024). Reaktualisasi Peran Mahasiswa sebagai Agen Perubahan dan Pemberdayaan dalam Masyarakat. *The Journalish: Social and Government*, 5(4), 53-542. <https://doi.org/10.55314/tsg.v5i4.913>

Suryanta, A., Widiyanti, L. W., & Ashari, M. (2025). PENGUJIAN WHITE BOX TERHADAP SISTEM INFORMASI AKADEMIK (SIKAD) DI POLITEKNIK ANGKATAN DARAT. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(2), 2384-2390. <https://doi.org/10.36040/jati.v9i2.12911>

-
- Sylvia, R. (2025). Penggunaan AI Berbasis Large Language Models (LLM) sebagai Media Interaktif dalam Pendidikan Bahasa dan Hukum di Perguruan Tinggi. Disiplin: Majalah Civitas Akademika Sekolah Tinggi Ilmu Hukum Sumpah Pemuda, 31(4).
- Talaohu, S. A., Soekarta, R., & Surahmanto, M. (2025). Implementasi LLM Pada Chatbot PMB Universitas Muhammadiyah Sorong Menggunakan Metode RAG Berbasis Website. Framework: Jurnal Ilmu Komputer dan Informatika, 3(2). <https://doi.org/10.33506/framework.v3i02.4790>
- Wicaksono, A. B., & Suryawan, S. H. (2023). Implementasi Metode Rapid Application Development Pada Pengembangan Perpustakaan Digital Web di Rumah Sakit Jiwa Atma Husada Mahakam. Nanggroe: Jurnal Pengabdian Cendikia, 2(9).