



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**RANCANG BANGUN *CHATBOT* MENGGUNAKAN
LARGE LANGUAGE MODEL SEBAGAI SARANA
INFORMASI SKRIPSI**

SKRIPSI

VIDI SEPTRI ARGALUS MP 2107411032

**POLITEKNIK
NEGERI
JAKARTA**

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN
KOMPUTER
DEPOK
2025**



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



RANCANG BANGUN *CHATBOT* MENGGUNAKAN *LARGE LANGUAGE MODEL* SEBAGAI SARANA INFORMASI SKRIPSI

SKRIPSI

Dibuat untuk Melengkapi Syarat-Syarat yang Diperlukan untuk
Memperoleh Diploma Empat Politeknik

VIDI SEPTRI ARGALUS MP 2107411032

POLITEKNIK
NEGERI
JAKARTA

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN
KOMPUTER

DEPOK

2025



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan di bawah ini:

Nama : Vidi Septri Argalus Mp

NIM : 2107411032

Jurusan/Program Studi : Teknik Informatika dan Komputer / Teknik Informatika

Judul Skripsi : RANCANG BANGUN CHATBOT MENGGUNAKAN
LARGE LANGUAGE MODEL SEBAGAI SARANA
INFORMASI SKRIPSI

Menyatakan dengan sebenarnya bahwa skripsi ini benar-benar merupakan hasil karya saya sendiri, bebas dari peniruan terhadap karya dari orang lain. Kutipan pendapat dan tulisan orang lain ditunjuk sesuai dengan cara-cara penulisan karya ilmiah yang berlaku.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa dalam skripsi ini terkandung ciri-ciri plagiat dan bentuk-bentuk peniruan lain yang dianggap melanggar peraturan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

**POLITEKNIK
NEGERI
JAKARTA**

Depok, 3 Juli 2025

Yang membuat pernyataan,



Vidi Septri Argalus Mp

NIM 2107411032



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LEMBAR PENGESAHAN

Skripsi diajukan oleh:

Nama : Vidi Septri Argalus Mp
NIM : 2107411032
Program Studi : Teknik Informatika
Judul Skripsi : RANCANG BANGUN CHATBOT MENGGUNAKAN
LARGE LANGUAGE MODEL SEBAGAI SARANA
INFORMASI SKRIPSI

Telah diuji oleh tim penguji dalam sidang skripsi pada hari Rabu, 25 Juni 2025 dan dinyatakan LULUS.

Disahkan oleh:

Pembimbing I : Asep Kurniawan, S.Pd., M.Kom. (Asep Kurniawan)

Penguji I : Euis Oktavianti, S.Si., M.Ti. (Euis Oktavianti)

Penguji II : Rizki Elisa Nalawati, S.T., M.T. (Rizki Elisa Nalawati)

Penguji III : Fachroni Arbi Murad, S.Kom., M.Kom. (Fachroni Arbi Murad)

Mengetahui:

Jurusan Teknik Informatika dan Komputer

Ketua Jurusan

Anita Hidayati, S.Kom., M.Kom.
NIP-197908032003122003



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Politeknik Negeri Jakarta, saya yang bertandatangan di bawah ini:

Nama : Vidi Septri Argalus Mp

NIM : 2107411032

Jurusan/Program Studi : Teknik Informatika dan Komputer / Teknik Informatika

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Politeknik Negeri Jakarta Hak Bebas Royalti Non-Eksklusif atas karya ilmiah saya yang berjudul:

RANCANG BANGUN *CHATBOT* MENGGUNAKAN *LARGE LANGUAGE MODEL* SEBAGAI SARANA INFORMASI SKRIPSI

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini Politeknik Negeri Jakarta berhak menyimpan, mengalihmediakan/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenar-benarnya.

Depok, 3 Juli 2025

Yang menyatakan,


Vidi Septri Argalus Mp
NIM 2107411032





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

KATA PENGANTAR

Puji dan Syukur kepada Tuhan Yang Maha Kuasa, penulis ucapkan karena telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Rancang Bangun Chatbot Menggunakan Large Language Model Sebagai Sarana Informasi Skripsi Jurusan Teknik Informatika dan Komputer Politeknik Negeri Jakarta” sebagai salah satu persyaratan untuk mencapai gelar Sarjana Terapan Program Studi Teknik Informatika di Politeknik Negeri Jakarta.

Banyak pihak yang telah memberikan dukungan dan bantuan dalam proses penyelesaian studi dan skripsi ini, sehingga sudah sepantasnya penulis mengungkapkan rasa terima kasih dan berdoa agar Tuhan memberikan balasan kepada mereka yang telah mendukung penulis. Secara khusus penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Ibu Dr. Anita Hidayati, S.Kom., M.Kom., selaku Ketua Jurusan Teknik Informatika dan Komputer.
2. Ibu Euis Oktavianti, S.Si., M.Ti., selaku Kepala Program Studi Teknik Informatika.
3. Bapak Asep Kurniawan, S.Pd., M.Kom., selaku dosen pembimbing yang telah sabar membimbing, mengarahkan dan membantu dalam proses penyelesaian skripsi ini.
4. Seluruh Bapak/Ibu dosen yang sudah mendidik penulis sehingga menjadi pribadi yang lebih baik.
5. Kedua orang tua penulis yang mendoakan, mendukung dan memberikan kasih sayang kepada penulis.
6. Saudara kandung penulis yang memberikan dukungan dan motivasi dalam menyusun skripsi.



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

7. Teman-teman penulis yang selalu memberikan semangat untuk menyelesaikan skripsi penulis.

Akhir kata, penulis berdoa agar skripsi ini bermanfaat bagi banyak pihak, baik untuk pendidikan maupun penelitian yang akan datang. Penulis juga sadar bahwa skripsi ini jauh dari kata sempurna sehingga mengucapkan permohonan maaf atas keterbatasan dalam skripsi ini.

Depok, 11 Juni 2025

Penulis,

Vidi Septri Argalus Mp

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

RANCANG BANGUN *CHATBOT* MENGGUNAKAN *LARGE LANGUAGE MODEL* SEBAGAI SARANA INFORMASI SKRIPSI

ABSTRAK

Mahasiswa Jurusan Teknik Informatika dan Komputer Politeknik Negeri Jakarta seringkali menghadapi kesulitan dalam memperoleh informasi terkait skripsi yang menyebabkan inefisiensi layanan akademik. Penelitian ini bertujuan untuk merancang sistem chatbot berbasis *Large Language Model* (LLM) dengan pendekatan *Retrieval-Augmented Generation* (RAG) sebagai sarana informasi skripsi yang interaktif dan efisien. Sistem ini diwujudkan melalui proses pengembangan yang iteratif dan adaptif, dengan sistem yang dibangun menggunakan teknologi seperti FastAPI untuk *backend*, ReactJS untuk *frontend*, dan Pinecone sebagai *vector database*. Evaluasi sistem dilakukan melalui serangkaian pengujian komprehensif yang meliputi *Black Box Testing*, *User Acceptance Test* (UAT) yang melibatkan 10 mahasiswa dan 1 admin, *System Usability Scale* (SUS), dan *Net Promoter Score* (NPS). Hasil pengujian menunjukkan fungsionalitas sistem mencapai 100% efektif berdasarkan *Black Box Testing*. Skor UAT dari mahasiswa mencapai 92,6% (Sangat Baik), dan dari admin mencapai 100% (Sangat Efektif). Sistem juga memperoleh skor kegunaan rata-rata SUS sebesar 78,67 (*Good*) dan skor NPS 54,16% (*Excellent*). Berdasarkan hasil tersebut, disimpulkan bahwa chatbot yang dikembangkan berhasil menjadi solusi yang fungsional, andal, dan diterima dengan sangat baik oleh pengguna untuk meningkatkan aksesibilitas informasi skripsi.

Kata Kunci: *Chatbot*, Informasi Skripsi, *Large Language Model*, *Retrieval-Augmented Generation*, *Web*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR ISI

PERNYATAAN ORISINALITAS.....	ii
HALAMAN PENGESAHAN.....	iii
PERSETUJUAN PUBLIKASI.....	iv
KATA PENGANTAR.....	v
ABSTRAK.....	vii
DAFTAR ISI.....	viii
DAFTAR TABLE	x
DAFTAR GAMBAR.....	xi
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah.....	2
1.3. Batasan Masalah.....	2
1.4. Tujuan dan Manfaat.....	3
1.5. Sistematika Penulisan.....	3
BAB II TINJAUAN PUSTAKA.....	5
2.1 Landasan Teori	5
2.1.1 Chatbot.....	5
2.1.2 Large Language Model.....	5
2.1.3 LangChain.....	6
2.1.4 Groq Cloud.....	6
2.1.5 OpenAI.....	6
2.1.6 FastAPI.....	7
2.1.7 Visual Studio Code.....	7
2.1.8 ReactJS.....	7
2.1.9 RAG.....	8
2.1.10 PDF.....	9
2.1.11 Pinecone.....	9
2.1.12 Mistral AI.....	10
2.1.13 Digital Ocean.....	10
2.1.14 Unified Modeling Language (UML).....	11
2.1.15 Activity Diagram.....	11
2.2 Penelitian Terdahulu.....	13
BAB III METODE PENELITIAN	19
3.1 Rancangan Penelitian	19



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3.2	Tahapan Penelitian	19
3.3	Obyek Penelitian	20
BAB IV HASIL DAN PEMBAHASAN		21
4.1	Identifikasi Kebutuhan	21
4.1.1	Kebutuhan Fungsional	21
4.1.2	Kebutuhan Non-Fungsional	23
4.2	Perancangan Sistem	25
4.2.1	Deskripsi Aplikasi	25
4.2.2	Diagram UML	27
4.2.3	Arsitektur Sistem	31
4.3	Implementasi Sistem	40
4.3.1	Lingkungan Implementasi	41
4.3.2	Implementasi <i>Backend</i>	41
4.4	Pengujian Sistem	65
4.4.1	Deskripsi Pengujian	65
4.4.2	Hasil Pengujian	70
BAB V PENUTUP		80
5.1	Kesimpulan	80
5.2	Saran	80
DAFTAR PUSTAKA		82
DAFTAR RIWAYAT HIDUP PENULIS		85
LAMPIRAN		86
Lampiran 1 Hasil Blackbox Testing		86
Lampiran 2 UAT dengan Admin Program Studi TI TIK		87
Lampiran 3 Data Referensi Evaluasi RAG dan <i>Response Chatbot</i>		88



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR TABLE

Tabel 2.1	Perbandingan penelitian terdahulu	14
Tabel 4.1	Kebutuhan Fungsional	21
Tabel 4.2	Kebutuhan Non-Fungsional.....	23
Tabel 4.3	Kebutuhan Hardware dan Software.....	24
Tabel 4.4	<i>Endpoint</i> API	43
Tabel 4.5	Sistem Middleware	45
Tabel 4.6	Pengujian Blackbox Testing	66
Tabel 4.7	Kriteria UAT	67
Tabel 4.8	Pertanyaan UAT User Mahasiswa Sistem <i>Chatbot</i>	68
Tabel 4.9	Pengujian Fungsional Dengan Admin TI TIK	68
Tabel 4.10	Pertanyaan System Usability Scale.....	69
Tabel 4.11	Pertanyaan Net Promoter Score	70
Tabel 4.12	Hasil Skor RAGAS	71
Tabel 4.13	Hasil Skor ROUGE	71
Tabel 4.14	Hasil Pengujian Black Box	72
Tabel 4.15	Hasil User Acceptance Mahasiswa	73
Tabel 4.16	Hasil User Acceptance Admin	75
Tabel 4.17	Hasil SUS	76
Tabel 4.18	Hasil SUS 2	77
Tabel 4.19	Hasil Perhitungan SUS.....	77
Tabel 4.20	Hasil Perhitungan NPS.....	79



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR GAMBAR

Gambar 4.1	Alur Diagram Questions and Answer Chatbot Mahasiswa.....	26
Gambar 4.3	Use Case Diagram Sistem.....	27
Gambar 4.4	Activity Diagram Pendaftaran Akun.....	28
Gambar 4.5	Activity Diagram Login	29
Gambar 4.6	Activity Diagram Upload Files	30
Gambar 4.7	Activity Diagram Interaksi Chat	31
Gambar 4.8	Arsitektur Umum Sistem	32
Gambar 4.9	Arsitektur <i>Frontend</i>	33
Gambar 4.10	Arsitektur <i>Backend</i>	34
Gambar 4.11	Hubungan Sistem	36
Gambar 4.12	Benchmark Model Llama 3.....	37
Gambar 4.13	Diagram Penyimpanan Vector	39
Gambar 4.14	Struktur Proyek dan Modul <i>Backend</i>	41
Gambar 4.15	Implementasi Autentikasi	47
Gambar 4.16	Implementasi Retrieve Context kedalam Model.....	48
Gambar 4.17	Pengambilan <i>Chunk</i> yang Relevan	49
Gambar 4.18	Template Prompt.....	51
Gambar 4.19	Ekstrak Document.....	52
Gambar 4.20	Ekstrak Audio	53
Gambar 4.21	Segmentasi Teks.....	53
Gambar 4.22	Konfigurasi Nginx.....	55
Gambar 4.23	Struktur Folder	56
Gambar 4.24	Implementasi AuthProvider	58
Gambar 4.25	Implementasi Send Message <i>Request</i>	59
Gambar 4.26	Tampilan Interaksi <i>Chatbot</i>	60
Gambar 4.27	Manajemen Percakapan	60
Gambar 4.28	<i>Sources</i> Attribution	61
Gambar 4.29	Fitur copy to clipboard	62
Gambar 4.30	Pengunggahan <i>Sources</i>	62
Gambar 4.31	Mengambil <i>KnowledgeSourceList</i>	63
Gambar 4.31	Interpretasi Skor SUS.....	70
Gambar 4.32	Rumus Perhitungan Pengujian BlackBox	73



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB I PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi telah membawa perubahan signifikan dalam cara institusi pendidikan mengelola dan menyampaikan informasi. Saat ini, mahasiswa membutuhkan akses informasi yang cepat dan akurat terkait proses administrasi, khususnya yang berkaitan dengan tugas akhir seperti skripsi. Akan tetapi, sering kali mahasiswa menghadapi berbagai hambatan dalam mendapatkan informasi yang diperlukan. Dari 50 responden mahasiswa tingkat akhir, 45 mahasiswa mengatakan bahwa setiap dari mereka mengalami kesulitan dalam mendapatkan informasi skripsi, mahasiswa harus mengunjungi staf administrasi secara langsung untuk mendapatkan jawaban, yang dapat menyebabkan waktu terbuang dan antrean panjang. Selain itu, dalam pengalaman responden yang mendapatkan informasi melalui whatsapp memungkinkan informasi tertimbun dengan pesan terbaru, lalu informasi yang sering berubah, seperti jadwal pendaftaran dan dokumen yang harus dilengkapi, membuat mahasiswa kesulitan dalam memperoleh pembaruan terkini.

Di tengah permasalahan ini, kebutuhan akan sistem yang lebih responsif dan efisien menjadi semakin jelas. Pengelolaan informasi yang efektif di lingkungan pendidikan dapat berkontribusi pada pengalaman belajar yang lebih baik dan meningkatkan kepuasan layanan bagi mahasiswa. Salah satu solusi untuk menghadapi tantangan ini adalah melalui penggunaan teknologi *chatbot* berbasis *Large Language Models* (LLM) menggunakan Llama 3.3 yang dapat menyediakan layanan informasi otomatis dan interaktif (Dybedokken & Nilsen, 2024).

Chatbot merupakan sistem yang mampu berinteraksi dengan pengguna dalam bahasa alami dan memberikan respon sesuai konteks yang dibutuhkan. Hal ini sejalan dengan penelitian (Alexander Tobias, 2024) yang menunjukkan bahwa *user acceptance* tesing dengan 46 siswa, menemukan akurasi tinggi (88%) dan penerimaan positif dari siswa. *Chatbot* dapat meningkatkan efektivitas layanan informasi di sektor pendidikan, dengan kemampuan untuk menyediakan jawaban yang cepat dan mendetail (Solih, 2024).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Chatbot ini tidak hanya dapat memberikan jawaban atas pertanyaan terkait skripsi, tetapi juga memungkinkan pembaruan informasi secara dinamis melalui admin page. Dengan membangun *Chatbot* ini berbasis *web* diharapkan dapat meningkatkan efisiensi layanan administrasi khususnya pada lingkup Jurusan Teknik Informatika dan Komputer di Politeknik Negeri Jakarta.

Menghadapi berbagai tantangan dalam penelitian, institusi pendidikan perlu mengadopsi pendekatan inovatif yang melibatkan teknologi untuk merespon perkembangan zaman dan memenuhi harapan mahasiswa. Implementasi *chatbot* sebagai sistem informasi otomatis dapat menjadi langkah strategis untuk meningkatkan pelayanan administrasi, mempermudah akses informasi, dan menyediakan pengalaman belajar yang lebih baik (Assayed, Alkhatib, & Shaalan, 2024).

Dengan demikian, penelitian ini bertujuan untuk mengembangkan sebuah sistem *chatbot* untuk meringankan beban administrasi, sekaligus meningkatkan aksesibilitas dan efisiensi informasi bagi mahasiswa di Jurusan Teknik Informatika dan Komputer di Politeknik Negeri Jakarta.

1.2. Perumusan Masalah

Berdasarkan latar belakang tersebut, permasalahan yang dapat diangkat dalam penelitian ini adalah bagaimana merancang dan membangun chatbot berbasis web yang memanfaatkan Large Language Model (LLM) untuk memberikan layanan informasi terkait skripsi ?

1.3. Batasan Masalah

Agar penelitian ini lebih terfokus, batasan masalah dalam penelitian ini adalah merancang dan membangun chatbot berbasis web yang memanfaatkan Large Language Model (LLM) untuk memberikan layanan informasi terkait skripsi lingkup Jurusan Teknik Informatika dan Komputer Politeknik Negeri Jakarta,



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1.4. Tujuan dan Manfaat

Tujuan dari penelitian ini antara merancang dan membangun chatbot berbasis web yang memanfaatkan Large Language Model (LLM) untuk memberikan layanan informasi terkait skripsi lingkup Jurusan Teknik Informatika dan Komputer Politeknik Negeri Jakarta.

Manfaat dari penelitian ini:

1. Mempercepat pencarian informasi terkait skripsi dengan cepat tanpa harus bergantung pada jam kerja staf administrasi
2. Mengurangi beban kerja staf administrasi dalam menjawab pertanyaan rutin dari mahasiswa
3. Memberikan inovasi teknologi yang meningkatkan reputasi kampus dalam penerapan teknologi digital

1.5. Sistematika Penulisan

Untuk memudahkan penelusuran pembahasan yang ada pada penulisan ini, maka adapun penulisan laporan skripsi menggunakan sistematika sebagai berikut:

1. Bagian awal skripsi
Ini memuat halaman awal seperti halaman sampul, halaman judul, pernyataan orisinalitas, pengesahan, kata pengantar, abstrak, daftar isi, daftar tabel, daftar gambar dan daftar lampiran.

2. Bagian utama skripsi

Bagian ini terdiri dari bab-bab antara lain sebagai berikut:

- 1) Bab 1 Pendahuluan, yang berisi latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat serta sistematika penulisan.
- 2) Bab 2 Tinjauan Pustaka, yang meliputi hasil telaah penulis berdasarkan penelitian terdahulu yang memiliki topik berkaitan dengan penelitian yang dilakukan. Dilanjutkan dengan landasan teori yang berisi tentang pembahasan yang dilakukan pada penelitian.
- 3) Bab 3 Metode Penelitian, untuk menjabarkan tentang metode-metode penelitian yang digunakan dalam pengembangan yang



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

meliputi berbagai aspek, seperti pemilihan lokasi, waktu penelitian, analisis kebutuhan serta alur penelitian.

- 4) Bab 4 Hasil dan Pembahasan, terdiri dari hasil penelitian dan analisa yang dilakukan baik secara kualitatif, kuantitatif maupun kombinasi serta hasil penelitian.
- 5) Bab 5 Penutup, bab yang berisi kesimpulan dan saran berdasarkan penelitian. Disini menjabarkan kesimpulan yang dapat dikemukakan, kelemahan yang ada dan bagaimana keselarasan dengan tujuan dan manfaat penelitian. Dilanjutkan dengan saran yang dapat dilakukan untuk mengatasi masalah dan kelemahan yang ada.
3. Bagian Akhir Skripsi
Bagian akhir ini berisi tentang daftar pustaka serta daftar lampiran pendukung selama proses penelitian.

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Chatbot

Chatbot adalah program komputer yang dirancang untuk mensimulasikan percakapan manusia melalui teks. *Chatbot* digunakan dalam berbagai aplikasi seperti layanan pelanggan, pendidikan, hingga *e-commerce*, karena kemampuannya untuk memberikan respons yang cepat dan otomatis. Menurut *Oracle*, *chatbot* didefinisikan sebagai agen perangkat lunak berbasis kecerdasan buatan (*Artificial Intelligence*) yang dapat berinteraksi dengan manusia menggunakan bahasa alami (Oracle, October 7, 2020). *Chatbot* biasanya dirancang untuk:

- a. Menyelesaikan tugas spesifik: Seperti menjawab pertanyaan pelanggan.
- b. Berinteraksi secara kontekstual: Memahami maksud pengguna (*user intent*).
- c. Memproses dan merespons input: Baik melalui teks maupun suara.

Chatbot dapat diklasifikasikan menjadi dua jenis utama:

- a. *Rule-Based Chatbot*: Menggunakan aturan tetap untuk menentukan respons. Model ini bersifat terbatas karena tidak dapat menangani input yang tidak terdefinisi dalam aturannya.
- b. *AI-Powered Chatbot*: Didukung oleh teknologi machine learning dan NLP, sehingga dapat memahami input yang lebih kompleks dan menghasilkan respons yang kontekstual.

Dalam proyek ini, *chatbot* yang dikembangkan termasuk kategori *AI-Powered Chatbot*, dengan dukungan model *Large Language Model (LLM)* untuk memahami konteks percakapan dan menghasilkan respons berkualitas tinggi.

2.1.2 Large Language Model

Large Language Model (LLM) adalah jenis kecerdasan buatan (AI) yang didesain untuk memahami, menghasilkan, dan memanipulasi bahasa manusia dalam skala yang sangat besar. Bayangkan sebuah komputer yang bisa memahami dan merespon bahasa manusia layaknya seorang manusia. LLM dilatih dengan menggunakan data teks dalam jumlah yang sangat besar, sehingga model ini



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

mampu mempelajari pola, konteks, dan makna bahasa yang kompleks (Google, 2024).

2.1.3 LangChain

LangChain adalah *framework* yang dirancang untuk pengembangan aplikasi berbasis *Large Language Model (LLM)* seperti OpenAI GPT atau model lainnya. LangChain memungkinkan pengembang untuk mengintegrasikan komponen NLP secara modular dan efisien (LangChain, 2023). LangChain memiliki beberapa fitur unggulan:

- a. *Pipeline Modular*: Memisahkan proses menjadi modul yang dapat disesuaikan, seperti parsing data, pelatihan model, dan pembuatan respons.
- b. *Integrasi LLM*: Mendukung berbagai model *LLM* untuk pemrosesan bahasa alami.
- c. *Pemrosesan Dokumen*: Memudahkan pencarian informasi dari dokumen besar, seperti laporan atau artikel.

2.1.4 Groq Cloud

Groq *Cloud* adalah *platform cloud* yang dirancang untuk pengembangan dan pelatihan model AI, termasuk *Large Language Model (LLM)* (GroqCloud, 2023). Beberapa keunggulan Groq Cloud adalah:

- a. Kecepatan pemrosesan tinggi: Mendukung pelatihan model dengan efisiensi tinggi
- b. Kompatibilitas luas: Dapat digunakan dengan berbagai *framework*

2.1.5 OpenAI

OpenAI adalah *platform* yang menyediakan berbagai kemudahan dalam melakukan pengembangan produk berbasis AI (OpenAI, 2024). Embedding merupakan salah satu kemudahan yang ditawarkan oleh OpenAI. Embedding ini digunakan untuk mengolah eksternal informasi yang diterima oleh model LLM agar dapat diproses. Dimana setiap kata atau kalimat akan diwakili oleh sebuah *vector* yang mana *vector*-vektor ini ditata agar setiap kata yang memiliki makna serupa akan lebih dekat satu sama lain pada ruang *vector*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2.1.6 FastAPI

FastAPI adalah *framework* Python *modern* untuk pengembangan *Application Programming Interface* (API) yang cepat, efisien dan mudah digunakan (FastAPI, 2024). Penggunaan FastAPI dalam proyek untuk mnghubungkan *chatbot* dengan *platform* yang akan menerima data dari *backend* dan mengelola komunikasi antara komponen *backend* dan *frontend* secara efisien.

2.1.7 Visual Studio Code

Visual Studio Code atau yang sering disingkat dengan VS Code adalah *code editor* yang dikembangkan oleh Microsoft. VS Code ini bersifat gratis dan dapat digunakan pada berbagai sistem operasi seperti windows, macOS dan Linux. VS Code mendukung berbagai Bahasa pemograman serta ekstensi, intelisense yang membantu pengembangan proyek (Visual Studio Code, 2022). Beberapa fitur utama dari Visual Studio Code meliputi:

1. Editor yang Kuat: VS Code menyediakan fitur-fitur canggih seperti IntelliSense, yang memberikan saran kode berdasarkan konteks, serta debugging yang terintegrasi.
2. Ekstensi yang Beragam: Pengguna dapat menambahkan berbagai ekstensi untuk mendukung bahasa pemrograman atau alat pengembangan lainnya, sehingga meningkatkan fungsionalitas editor.
3. Kontrol Versi Git: VS Code memiliki integrasi bawaan dengan Git, memungkinkan pengguna untuk mengelola repositori kode mereka langsung dari editor.
4. Kustomisasi: Pengguna dapat menyesuaikan tema, pintasan keyboard, dan preferensi lainnya sesuai kebutuhan mereka.

2.1.8 ReactJS

React dalah sebuah library JavaScript yang dikembangkan oleh Facebook untuk membangun antarmuka pengguna (UI) (React, 2023). Berikut adalah beberapa poin penting tentang ReactJS:



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1. **Komponen:** ReactJS menggunakan konsep komponen, yaitu potongan kode yang dapat digunakan kembali untuk membangun elemen UI. Setiap komponen dapat memiliki state dan properti sendiri, yang membuatnya mudah untuk dikelola dan diubah.
2. **JSX:** React menggunakan JSX (JavaScript XML), yang memungkinkan pengembang untuk menulis kode HTML di dalam JavaScript. Ini mempermudah pembuatan elemen UI yang kompleks dengan sintaks yang lebih intuitif.
3. **Virtual DOM:** Salah satu fitur utama React adalah penggunaan Virtual DOM. Virtual DOM adalah representasi virtual dari DOM asli yang disimpan dalam memori. Ketika ada perubahan pada UI, React hanya memperbarui bagian yang berubah, bukan seluruh halaman, sehingga meningkatkan performa aplikasi.
4. **Unidirectional Data Flow:** React menerapkan aliran data satu arah, yang berarti data mengalir dari komponen induk ke komponen anak. Ini membantu dalam melacak perubahan data dan membuat aplikasi lebih mudah untuk di-debug.
5. **SEO-Friendly:** React dapat digunakan untuk membuat aplikasi yang ramah SEO karena memungkinkan rendering di sisi *server*, yang membantu mesin pencari untuk mengindeks halaman dengan lebih baik.

2.1.9 RAG

Retrieval-Augmented Generation (RAG) adalah pendekatan dalam kecerdasan buatan yang menggabungkan kemampuan model bahasa besar dengan mekanisme pengambilan informasi dari sumber data eksternal. Dalam sistem RAG, model tidak hanya mengandalkan pengetahuan yang diperoleh selama pelatihan, tetapi juga dapat mengambil data relevan dari *database*, dokumen, atau sumber informasi lainnya untuk menghasilkan respons yang lebih akurat dan kontekstual. Pendekatan ini memecahkan masalah utama model bahasa tradisional yang sering terbatas pada informasi yang sudah lama atau tidak lengkap. Dengan mengintegrasikan pengambilan data eksternal, RAG memungkinkan aplikasi AI, seperti *chatbot* dan sistem penjawab pertanyaan, untuk memberikan jawaban yang lebih tepat dan terkini. Selain itu, RAG dapat meningkatkan kemampuan AI dalam meringkas teks



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

dan membuat konten yang lebih relevan dengan kebutuhan pengguna. (ms-johnalex, 2025)

2.1.10 PDF

PDF, atau *Portable Document Format*, adalah format berkas yang dikembangkan oleh Adobe Systems pada tahun 1993 (Adobe, n.d.). Tujuan utama dari penciptaan format ini adalah untuk memungkinkan pertukaran dokumen digital yang dapat ditampilkan dan dicetak secara konsisten di berbagai platform perangkat keras dan perangkat lunak. PDF mampu merepresentasikan dokumen dua dimensi yang mencakup teks, huruf, citra, dan grafik vektor. Keunggulan utama PDF terletak pada kemampuannya untuk mempertahankan format asli dokumen, sehingga tata letak, font, dan gambar tetap sama terlepas dari perangkat yang digunakan untuk membukanya. Selain itu, PDF juga mendukung fitur keamanan seperti enkripsi dan tanda tangan digital, menjadikannya pilihan yang aman untuk berbagi dokumen penting. Dengan demikian, PDF telah menjadi standar global dalam pertukaran dan pengarsipan dokumen elektronik, digunakan secara luas dalam berbagai bidang seperti bisnis, pendidikan, dan pemerintahan.

2.1.11 Pinecone

Pinecone adalah layanan *vector database* yang dirancang khusus untuk menangani pencarian berbasis vektor (*vector search*). Teknologi ini sangat berguna dalam pengembangan aplikasi yang melibatkan kecerdasan buatan (AI) dan pemrosesan bahasa alami (Natural Language Processing/NLP), seperti *chatbot*, sistem rekomendasi, dan pencarian semantik (Pinecone, 2024).

Dalam konteks *Retrieval-Augmented Generation* (RAG), Pinecone digunakan untuk menyimpan dan mengelola representasi vektorial dari potongan teks (*chunk*). Vektor-vektor ini dihasilkan melalui proses *embedding* menggunakan model bahasa, dan memungkinkan sistem untuk melakukan pencarian berdasarkan kemiripan makna (*semantic similarity*) alih-alih pencocokan kata kunci literal.

Pinecone menawarkan keunggulan berupa:

- 1) Skalabilitas tinggi untuk menyimpan jutaan vektor.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- 2) Latensi rendah untuk pencarian cepat.
- 3) Pengelolaan data vektor secara otomatis dan efisien.
- 4) Dukungan penuh untuk *filtering* dan metadata.

Dengan menggunakan Pinecone, pengembang dapat membuat sistem pencarian yang lebih cerdas dan relevan, yang sangat bermanfaat dalam integrasi dengan sistem AI *modern*.

2.1.12 Mistral AI

Mistral AI adalah perusahaan teknologi asal Prancis yang mengembangkan model-model bahasa (*large language models*, LLM). Salah satu keunggulan dari model yang dikembangkan Mistral AI adalah performa tinggi dengan ukuran model yang relatif kecil dibandingkan dengan model sejenis lainnya, sehingga cocok digunakan pada berbagai skenario termasuk di perangkat lokal (*on-premise*) (Mistral, 2024). Mistral AI dikenal melalui rilis model seperti Mistral 7B dan Mixtral, yang mendukung berbagai tugas NLP seperti:

- 1) *Text summarization*
- 2) *Question answering*
- 3) *Document understanding*
- 4) *Text generation*

Dalam implementasi sistem RAG, model dari Mistral AI dapat digunakan untuk melakukan *optical character recognition* (OCR) dan ekstraksi informasi dari dokumen digital, seperti PDF atau gambar teks. Mistral OCR, sebagai bagian dari pipeline, mampu mengenali dan mengubah konten visual menjadi teks yang dapat diindeks dan diproses lebih lanjut oleh sistem.

2.1.13 Digital Ocean

DigitalOcean merupakan penyedia layanan infrastruktur *cloud* yang berfokus pada kesederhanaan dan kemudahan penggunaan bagi *developer* (Digital Ocean, 2025). Didirikan pada tahun 2011, perusahaan ini telah tumbuh menjadi salah satu pemain utama dalam industri *cloud computing*, menawarkan berbagai layanan yang memfasilitasi pengembangan, *deployment*, dan *scaling* aplikasi di internet. Filosofi inti DigitalOcean adalah menyediakan *server virtual* (sering disebut sebagai



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

"Droplets") yang dapat disiapkan dengan cepat dan dikelola dengan antarmuka yang intuitif, memungkinkan pengembang untuk berfokus pada kode mereka daripada kerumitan manajemen infrastruktur.

2.1.14 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa standar yang digunakan untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan dari sebuah sistem. Dalam pengembangan, UML berfungsi sebagai alat komunikasi yang universal, memungkinkan tim pengembang, klien, dan pemangku kepentingan lainnya untuk memahami arsitektur dan perilaku sistem dengan jelas. Penggunaan UML dalam penelitian ini penting untuk memodelkan dan mendokumentasikan desain sistem *chatbot*, memastikan semua fungsionalitas dan interaksi tergambar dengan jelas sebelum proses implementasi (IBM, 2023).

2.1.15 Activity Diagram

Diagram adalah salah satu jenis diagram perilaku dalam UML yang digunakan untuk memodelkan aliran kerja (*workflow*) atau proses bisnis dalam suatu sistem. Diagram ini menggambarkan urutan langkah-langkah atau aktivitas yang dilakukan, termasuk percabangan, penggabungan, dan perulangan. *Activity Diagram* sangat mirip dengan flowchart tradisional, namun memiliki kemampuan yang lebih kaya untuk menggambarkan konkurensi (aktivitas yang berjalan secara bersamaan) dan *sinkronisasi* (LucidChart, 2021).

2.1.16 Use Case Diagram

Use Case Diagram adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memodelkan interaksi antara pengguna (*user* atau *aktor*) dengan sistem yang dikembangkan. Diagram ini menggambarkan fungsionalitas sistem dari sudut pandang pengguna dan menjelaskan bagaimana suatu aktor berinteraksi dengan fitur-fitur utama dalam sistem.

Use case diagram merepresentasikan kebutuhan fungsional sistem dengan cara yang sederhana namun jelas. Diagram ini membantu dalam proses komunikasi



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

antara pengembang sistem dan pihak pemangku kepentingan (*stakeholders*) dalam memahami ruang lingkup sistem serta fitur-fitur yang akan dibangun.

2.1.17 System Usability Scale

System Usability Scale (SUS) adalah sebuah metode evaluasi *usability* (kegunaan) yang dikembangkan oleh John Brooke pada tahun 1986. SUS merupakan kuesioner sederhana yang terdiri dari 10 pernyataan yang dirancang untuk mengukur persepsi pengguna terhadap kemudahan penggunaan sebuah sistem, aplikasi, atau produk. Kuesioner ini menggunakan skala Likert dengan lima poin (1 = sangat tidak setuju, 5 = sangat setuju), di mana setiap pernyataan secara bergantian bersifat positif dan negatif.

SUS memiliki kelebihan karena bersifat praktis, cepat digunakan, dan telah terbukti reliabel dalam berbagai konteks pengujian sistem. Sebagai panduan umum, skor SUS di atas 70 dianggap sebagai skor yang baik (Kurniawan, Nofriadi, & Nata, 2022).

2.1.18 Net Promoter Score

Net Promoter Score (NPS) adalah sebuah metrik yang digunakan untuk mengukur loyalitas pelanggan atau pengguna terhadap suatu produk, layanan, atau organisasi. Konsep ini diperkenalkan oleh Fred Reichheld pada tahun 2003 dan sejak saat itu telah digunakan secara luas oleh berbagai perusahaan dan organisasi di seluruh dunia.

Pertanyaan dijawab dengan skala dari 0 hingga 10. Berdasarkan jawabannya, responden dikategorikan ke dalam tiga kelompok:

- 1) Promoters (skor 9–10): Pengguna yang sangat puas dan kemungkinan besar akan merekomendasikan produk/layanan.
- 2) Passives (skor 7–8): Pengguna yang cukup puas tetapi tidak antusias, dan rentan berpindah ke produk pesaing.
- 3) Detractors (skor 0–6): Pengguna yang tidak puas dan berpotensi memberikan ulasan negatif.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Skor NPS dihitung dengan mengurangi persentase *Detractors* dari persentase *Promoters*. Hasil akhir berkisar dari -100 hingga +100. Semakin tinggi skor NPS, semakin tinggi pula tingkat loyalitas dan kepuasan pengguna terhadap produk atau layanan yang ditawarkan (Bunker, 2021).

2.1.19 Retrieval-Augmented Generation Assessment RAGAS

RAGAS menyediakan serangkaian metrik evaluasi yang dapat digunakan untuk mengukur performa aplikasi Large Language Model (LLM) secara objektif. Metrik-metrik ini dirancang khusus untuk membantu dalam menilai kinerja aplikasi (Ragas, 2025).

Setiap metrik adalah paradigma yang didesain untuk mengevaluasi aspek tertentu dari aplikasi. Metrik berbasis LLM mungkin memerlukan satu atau lebih panggilan LLM untuk menghasilkan skor atau hasilnya. Selain itu, pengguna juga memiliki fleksibilitas untuk memodifikasi atau membuat metrik sendiri menggunakan RAGAS sesuai dengan kebutuhan spesifik evaluasi mereka.

2.1.20 Recall-Oriented Understudy for Gisting Evaluation ROUGE

ROUGE memiliki seperangkat metrik yang banyak digunakan untuk mengevaluasi kualitas ringkasan teks otomatis dan terjemahan mesin. Nama "Recall-Oriented" menunjukkan fokus utamanya pada sejauh mana ringkasan yang dihasilkan oleh sistem atau jawaban sistem berhasil menangkap informasi penting dari ringkasan referensi (*ground truth*) yang dibuat oleh pengembang ketika ingin melakukan evaluasi (Otten, 2024).

2.2 Penelitian Terdahulu

Penelitian terdahulu adalah bagian dari tahap penelitian dalam mencari referensi serta membandingkan maupun melanjutkan kajian terdahulu. Bagian ini juga menunjukkan serta menghargai orisinalitas dari penelitian maupun yang terdahulu. Berikut merupakan penelitian terdahulu berdasarkan penelusuran penulis yang masih terkait dengan tema yang dikaji.



Tabel 2.1 Perbandingan penelitian terdahulu

No	Judul Penelitian	Author	Pembahasan	Persamaan / Perbedaan
1	Automated Information Retrieval and Services of Graduate School Using Chatbot System (2022)	Meennapa Rukhiran, Paniti Netinant	Menjelaskan tentang bagaimana melakukan otomatisasi dalam memberikan informasi sebagai pelayanan dalam bentuk sistem <i>chatbot</i> kepada murid-murid yang ada disekolah seperti informasi kelulusan murid. <i>Chatbot</i> terintegrasi dengan aplikasi Line dan GoogleForm, alurnya <i>chatbot</i> akan memberikan file pada opsi pertanyaan yang telah disediakan sistem.	Persamaan: sama-sama membangun <i>chatbot</i> untuk layanan akademik. Perbedaan: penelitian Meennapa dan paniti berbasis aturan (<i>rule-based</i>), sedangkan penelitian ini menggunakan <i>Large Language Model</i> (LLM).
2	Artificial Intelligence Chatbot Platform: AI Chatbot Platform for Educational Recommendations in Higher Education (2024)	Thanarat Kingchang, Pinanta Chatwattana, and Panita Wannapiroon	Artikel ini merinci pengembangan dan evaluasi <i>platform chatbot</i> AI yang dirancang untuk memberikan rekomendasi pendidikan kepada calon mahasiswa pendidikan tinggi. <i>Platform</i> ini, yang diakses melalui aplikasi Line, menggunakan algoritma random forest sederhana	Persamaan: sama-sama membangun <i>chatbot</i> . Perbedaan: pada penelitian terdahulu berfokus untuk memberikan saran kepada murid rekomendasi kuliah, juga menggunakan <i>platform</i> botnai sebagai tools development <i>chatbot</i> ,

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

			dan pemrosesan bahasa alami (NLP) untuk menganalisis bakat pengguna dan menawarkan saran program yang disesuaikan. Tujuan akhirnya adalah untuk menciptakan teknologi pendukung keputusan yang dapat diakses dengan mudah dan dipersonalisasi bagi calon mahasiswa dalam menjelajahi pilihan pendidikan tinggi.	sedangkan pada penelitian ini menggunakan <i>Large Language Models</i> (LLM).
3	An LLM-Driven Chatbot in Higher Education for Databases and Information Systems(2025)	Alexander Tobias Neumann , Yue Yin , Sulayman Sowe , Stefan Decker , and Matthias Jarke	Artikel penelitian ini meneliti efektivitas MoodleBot, sebuah chatbot berbasis model bahasa besar (LLM) sumber terbuka yang terintegrasi dalam sistem manajemen pembelajaran Moodle (LMS). Studi ini mengeksplorasi penerimaan siswa terhadap MoodleBot sebagai alat dukungan pembelajaran dan menilai akurasi responsnya terhadap konten kursus yang telah	Persamaan: sama-sama membangun chatbot menggunakan LLM Perbedaanya: variabel Y dimana pada penelitian terdahulu ini adalah efektifitas peningkatan pembelajaran murid, sedangkan pada penelitian saat ini variabel Y nya adalah efektifitas mendapatkan informasi skripsi pada Politeknik Negeri Jakarta jurusan TIK.



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

			ditetapkan. Menggunakan pendekatan Retrieve Augmented Generative (RAG) dan Technology Acceptance Model (TAM), para peneliti mengevaluasi kinerja MoodleBot dengan 46 siswa, menemukan akurasi tinggi (88%) dan penerimaan positif dari siswa. Temuan ini menyoroti potensi <i>chatbot</i> berbasis LLM untuk meningkatkan pembelajaran yang dipersonalisasi, mengurangi beban kerja guru, dan meningkatkan pembelajaran.	
4	Implementasi Azure Cognitive Services dalam Pengembangan <i>Chatbot</i> Layanan Informasi Skripsi (2022)	Nur Asiah Ramdani, Asriyanik, Winda Apriandari	Artikel penelitian ini merinci pengembangan dan pengujian <i>chatbot</i> untuk program Ilmu Komputer di Universitas Muhammadiyah Sukabumi. <i>Chatbot</i> ini, yang dibangun menggunakan Microsoft Azure Cognitive Services QnA Maker, bertujuan	Persamaan: sama-sama mengembangkan <i>chatbot</i> untuk kebutuhan sarana informasi skripsi kepada mahasiswa. Perbedaan: Pada <i>chatbot</i> penelitian terdahulu menerapkan konsep <i>intent-based</i> ataupun <i>key-value</i>, dimana



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

			untuk meningkatkan efisiensi dan efektivitas penyampaian informasi skripsi kepada mahasiswa. Para penulis menggunakan metodologi prototipe, melakukan pengujian alfa dan beta untuk menilai kinerja <i>chatbot</i> tersebut. Hasilnya menunjukkan bahwa <i>chatbot</i> berhasil memenuhi kebutuhan mahasiswa dengan memberikan pencarian informasi skripsi yang akurat, efisien, dan efektif, yang pada akhirnya menunjukkan kelayakan dan manfaat penggunaan Azure Cognitive Services dalam aplikasi pendidikan.	ditetapkannya fix skenario pertanyaan-jawaban.
5	<i>Chatbot Development for an Interactive Academic Information Services using the Rasa Open source Framework (2021)</i>	Dirko G. S. Ruindungan, Agustinus Jacobus	Artikel penelitian ini menjelaskan pengembangan <i>chatbot</i> untuk memberikan informasi administratif kepada mahasiswa	Persamaan: sama-sama membangun <i>chatbot</i> Perbedaan: pada penelitian terdahulu ini menggunakan Rasa <i>Open</i>



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

			universitas. Karena ketidakefisienan layanan informasi online yang ada dan beban pada sumber daya manusia akibat pertanyaan langsung dari mahasiswa melalui aplikasi obrolan, para peneliti membangun <i>chatbot</i> menggunakan kerangka kerja Rasa <i>Open source</i>. <i>Chatbot</i> ini dilatih dengan pertanyaan yang sering diajukan (FAQ), mencapai kinerja tinggi dalam Pemahaman Bahasa Alami (NLU) dengan presisi, recall, dan skor F1 mendekati 0,995, serta kinerja moderat dalam manajemen dialog dengan akurasi 0,70. Studi ini diakhiri dengan demonstrasi fungsi <i>chatbot</i> dan evaluasi kinerjanya.	<i>source</i>, serta <i>intent-based</i> dan juga variabel Y nya merujuk ke informasi administratif akademik sedangkan penelitian saat ini lebih berfokus pada skripsi. Dan juga menggunakan konsep RAG dalam mendapatkan informasi
--	--	--	---	--



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB III

METODE PENELITIAN

3.1 Rancangan Penelitian

Penelitian ini dirancang untuk mengembangkan sebuah *chatbot* berbasis *Large Language Model* (LLM) yang berfungsi sebagai sistem informasi untuk memberikan layanan terkait skripsi di Politeknik Negeri Jakarta, khususnya di Jurusan Teknik Informatika dan Komputer. Penelitian ini menggunakan pendekatan kuantitatif dan kualitatif, yang digunakan untuk menganalisis dan membangun aplikasi. Pendekatan ini dipilih karena penelitian ini bertujuan untuk mengembangkan *chatbot* yang dapat memahami bahasa manusia dan memberikan respon dengan akurasi tinggi. Pendekatan kuantitatif dibutuhkan untuk mengumpulkan data kuantitatif dari mahasiswa mengenai kebutuhan mereka dalam mengakses informasi terkait skripsi melalui kuesioner yang dibagikan, setelah itu pendekatan ini juga dipakai pada saat evaluasi sistem, yang menggunakan *System Usability Scale* dan *Net Promoter Score* untuk mengukur kegunaan sistem. Kemudian untuk pendekatan kualitatif nya akan melibatkan pengumpulan data deskriptif dari interaksi dan pengguna melalui wawancara ataupun observasi langsung nantinya selama *User Acceptance Test* (UAT). Selain itu juga menggali wawasan dari pengguna bagaimana saran pengembangan maupun ekspektasi mereka terhadap *chatbot*. Untuk metode yang digunakan adalah studi kasus dimana melakukan penelitian yang difokuskan untuk mengembangkan suatu aplikasi untuk dapat menjawab permasalahan yang sedang terjadi.

3.2 Tahapan Penelitian

Tahapan penelitian ini dirancang berdasarkan model *Large Language Model* yang meliputi beberapa langkah utama, antara lain:

1. Studi Literatur

Mengumpulkan referensi dan informasi terkait hal-hal yang menjadi permasalahan yang ingin diselesaikan. Penelusuran pustaka, jurnal, artikel, dan penelitian sebelumnya yang relevan dengan topik penelitian. Mengidentifikasi metode, teknik, dan teknologi terbaru yang dapat digunakan dalam penelitian ini.

2. Analisis Kebutuhan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada tahapan awal ini melakukan analisis kebutuhan, mengidentifikasi kebutuhan pengguna melalui survei menggunakan kuesioner untuk mengumpulkan data terkait kendala yang dihadapi pengguna dalam mengakses informasi skripsi.

3. Perancangan Sistem

Mendesain arsitektur *chatbot* dengan memanfaatkan *framework* LangChain untuk mengintegrasikan model LLM dengan *database* dinamis. Selanjutnya membuat mockup antarmuka admin page yang *user-friendly* untuk memudahkan pengelolaan data *chatbot*.

4. Implementasi Sistem

Pada tahapan ini mengembangkan *chatbot* dengan integrasi API LLM untuk menjawab pertanyaan terkait skripsi dan memberikan rekomendasi topik berdasarkan data pengguna.

5. Pengujian Sistem

Melakukan pengujian fungsional menggunakan skenario kasus uji untuk memastikan *chatbot* dapat menjawab pertanyaan dengan akurat, mengukur kinerja *chatbot* melalui analisa interaksi. Jika memungkinkan dapat melakukan *user acceptance testing* (UAT) untuk mengevaluasi kepuasan pengguna dengan mengumpulkan feedback secara langsung.

6. Evaluasi dan Dokumentasi

Mengumpulkan umpan balik dari pengguna untuk menyempurnakan prototipe *chatbot* agar lebih optimal.

3.3 Obyek Penelitian

Obyek penelitian dalam pengembangan *chatbot* ini meliputi:

1. Subyek penelitian, merupakan mahasiswa tingkat akhir, dosen pembimbing, dan admin skripsi di Jurusan TIK Politeknik Negeri Jakarta sebagai pengguna utama sistem.
2. Data penelitian, informasi terkait prosedur administrasi skripsi, jadwal seminar, format dokumen.
3. Lingkungan penelitian, sistem informasi skripsi yang ada di Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB IV

HASIL DAN PEMBAHASAN

4.1 Identifikasi Kebutuhan

Identifikasi kebutuhan menguraikan kebutuhan sistem *chatbot* yang dikembangkan, yang terdiri dari kebutuhan fungsional (fungsi-fungsi spesifik yang harus dimiliki sistem) dan kebutuhan non-fungsional. Identifikasi ini didasarkan pada latar belakang, perumusan masalah, batasan masalah, serta tujuan penelitian yang telah disampaikan dalam dokumen awal penelitian.

4.1.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah fitur atau fungsi spesifik yang harus dimiliki oleh *chatbot* agar dapat mencapai tujuan penelitian, yaitu menyediakan layanan informasi terkait skripsi secara otomatis dan akurat. Berikut adalah kebutuhan fungsional yang diidentifikasi dapat dilihat pada Tabel 4.1.

Tabel 4.1 Kebutuhan Fungsional

Kode	Kebutuhan Fungsional	Deskripsi
KF-1	Respon Otomatis Pertanyaan Skripsi	Sistem harus mampu menerima pertanyaan dari mahasiswa terkait informasi skripsi dan memberikan jawaban secara otomatis sesuai konteks. Hal ini penting untuk mengurangi beban pada staf administrasi dan mempercepat akses informasi.
KF-2	Integrasi dengan Website	Sistem harus terintegrasi dengan <i>website</i> sehingga mahasiswa dapat mengakses <i>chatbot</i> dengan mudah melalui antarmuka web yang responsif.
KF-3	Pembaharuan Informasi secara Dinamis	<i>Chatbot</i> harus dapat meng-update informasi melalui halaman admin (admin page) sehingga data yang disajikan selalu terkini. Ini mengakomodasi perubahan jadwal, format dokumen, dan ketentuan administrasi yang sering berubah.
KF-4	Memproses pertanyaan pengguna dalam bahasa alami	<i>Chatbot</i> harus mampu memahami dan menjawab pertanyaan pengguna dalam bahasa alami dengan respons yang akurat dan kontekstual.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

KF-5	Pengelolaan User dan Manajemen Log	Sistem perlu mencatat riwayat interaksi pengguna untuk evaluasi performa dan sebagai bahan analisis perbaikan ke depan, serta menyediakan fitur manajemen akun untuk admin dan pengguna.
------	------------------------------------	--

Setiap kebutuhan fungsional yang telah diidentifikasi memiliki peran penting dalam memastikan *chatbot* dapat berjalan sesuai dengan tujuan penelitian ini. Fitur respon otomatis terhadap pertanyaan skripsi menjadi aspek utama, mengingat mahasiswa sering menghadapi kendala dalam mendapatkan informasi administrasi secara cepat. Dengan kemampuan *chatbot* untuk merespons pertanyaan secara instan, beban kerja staf administrasi dapat dikurangi, sekaligus meningkatkan efisiensi layanan akademik. Selain itu, *chatbot* harus memiliki kemampuan pencarian dan penyajian informasi yang akurat dan dinamis, sehingga mahasiswa dapat memperoleh data terkini terkait prosedur administrasi, jadwal seminar, dan persyaratan dokumen tanpa harus mencari informasi secara manual atau menunggu balasan dari pihak kampus.

Agar dapat diakses dengan mudah oleh seluruh mahasiswa, *chatbot* harus terintegrasi dengan *website*. Dengan integrasi ini, mahasiswa tidak perlu menginstal aplikasi tambahan, melainkan cukup mengakses *chatbot* langsung melalui antarmuka web yang responsif. Selain itu, *chatbot* harus memiliki fitur pembaruan informasi secara dinamis yang memungkinkan admin untuk mengubah atau menambahkan data kapan saja melalui halaman admin. Hal ini sangat penting mengingat informasi akademik, seperti jadwal seminar dan ketentuan administrasi, sering mengalami perubahan yang perlu segera diinformasikan kepada mahasiswa. Selain itu, sistem juga harus mendukung pengelolaan user dan manajemen log guna mempermudah analisis performa *chatbot* serta meningkatkan pengalaman pengguna. Riwayat interaksi yang tercatat dapat digunakan untuk mengidentifikasi pola pertanyaan yang sering diajukan, sehingga *chatbot* dapat terus dikembangkan agar semakin akurat dalam memberikan jawaban. Dengan dukungan teknologi *Large Language Model* (LLM), *chatbot* diharapkan dapat memahami konteks pertanyaan mahasiswa dengan baik dan memberikan jawaban yang sesuai.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Keunggulan ini menjadi faktor kunci dalam memastikan bahwa *chatbot* tidak hanya sekedar menampilkan jawaban berdasarkan kata kunci, tetapi benar-benar memahami maksud pengguna dan memberikan respons yang relevan.

Dengan memenuhi kebutuhan fungsional ini, *chatbot* yang dikembangkan diharapkan mampu memberikan solusi efektif terhadap permasalahan akses informasi skripsi di Jurusan Teknik Informatika Politeknik Negeri Jakarta. Fitur-fitur yang dirancang bertujuan untuk meningkatkan efisiensi layanan akademik, mengurangi waktu tunggu mahasiswa dalam mendapatkan informasi, serta memastikan data yang disajikan selalu relevan dan terkini.

4.1.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional menjelaskan aspek kualitas dari sistem yang mendukung kinerja dan kehandalan layanan. Beberapa kebutuhan non-fungsional yang diidentifikasi dapat dilihat pada Tabel 4.2 :

Tabel 4.2 Kebutuhan Non-Fungsional

Kode	Kebutuhan Non-Fungsional	Deskripsi
KNF-1	Ketersediaan (Availability)	Sistem harus mampu memberikan respons setiap saat. ketersediaan tinggi secara tidak langsung mempengaruhi persepsi kinerja. Chatbot yang sering <i>down</i> tentu dianggap memiliki kinerja buruk.
KNF-2	Antarmuka Pengguna (<i>User Interface</i>)	Antarmuka harus user-friendly, intuitif, dan responsif sehingga memudahkan pengguna dalam mengakses fitur <i>chatbot</i> tanpa mengalami kebingungan.
KNF-3	Kompatibilitas	<i>Chatbot</i> harus dapat diakses melalui berbagai perangkat dan browser yang digunakan pengguna.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Selain kebutuhan fungsional, *chatbot* yang dikembangkan juga harus memenuhi berbagai kebutuhan non-fungsional guna memastikan kualitas, keandalan, dan kenyamanan pengguna dalam mengakses layanan informasi skripsi. Salah satu aspek utama adalah ketersediaan (*availability*), di mana *chatbot* harus mampu merespons pertanyaan pengguna dalam waktu kapansaja. Respons yang cepat akan meningkatkan pengalaman pengguna serta memastikan bahwa mahasiswa dapat memperoleh informasi secara instan tanpa hambatan waktu.

Dari sisi pengalaman pengguna, *chatbot* harus memiliki antarmuka pengguna (*user interface/UI*) yang intuitif, responsif, dan mudah digunakan oleh mahasiswa maupun admin. Tampilan yang user-friendly akan mempermudah mahasiswa dalam berinteraksi dengan *chatbot*, sementara admin dapat dengan mudah mengelola pembaruan informasi melalui halaman admin yang terorganisir dengan baik. Untuk memastikan sistem dapat terus dikembangkan dan diperbarui sesuai dengan kebutuhan pengguna. Selain itu, aspek kompatibilitas menjadi pertimbangan penting dalam pengembangan *chatbot* ini. Sistem harus dapat diakses melalui berbagai perangkat, baik *desktop*, *tablet*, maupun *smartphone*, agar mahasiswa dapat menggunakannya kapan saja dan di mana saja tanpa hambatan teknis. Dengan memastikan *chatbot* memenuhi semua kebutuhan non-fungsional ini, sistem yang dikembangkan akan memiliki performa optimal, mampu memberikan pengalaman pengguna yang nyaman, serta dapat diandalkan sebagai sarana informasi skripsi di Jurusan Teknik Informatika Politeknik Negeri Jakarta. Selanjutnya untuk menunjang kebutuhan pengembangan aplikasi, kebutuhan perangkat keras dan juga *software* meliputi:

Tabel 4.3 Kebutuhan Hardware dan Software

Kode	Kebutuhan	Deskripsi
KHS-1	CPU	Core 12 / Thread 16
KHS-2	RAM	16 GB
KHS-3	GPU	NVIDIA RTX 3050
KHS-4	Text Editor	Visual Studio Code
KHS-5	Bahasa Pemograman	Python, Javascript
KHS-6	Browser	Chrome
KHS-7	Framework <i>Backend</i>	FastAPI dan Langchain



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

KHS-8	Framework <i>Frontend</i>	NextJS
KHS-9	<i>Database</i>	PostgreSQL
KHS-10	Vector Store	Pinecone
KHS-11	Manajemen Dependensi	Pip dan NPM
KHS-12	Tools Pengembangan	Git dan Github
KHS-13	Deployment <i>Frontend</i>	Vercel
KHS-14	Deployment <i>Backend</i>	Digital Ocean

4.2 Perancangan Sistem

Pada bab ini dijelaskan secara komprehensif mengenai perancangan sistem *chatbot* berbasis *Large Language Model* (LLM) sebagai sarana informasi skripsi. Perancangan sistem mencakup beberapa aspek utama, yaitu arsitektur sistem, desain antarmuka pengguna (UI), desain *database*, diagram alur proses, dan penjelasan detail mengenai pemilihan desain serta teknologi yang digunakan. Setiap elemen perancangan ini dirancang agar dapat memenuhi kebutuhan fungsional dan non-fungsional yang telah diidentifikasi sebelumnya.

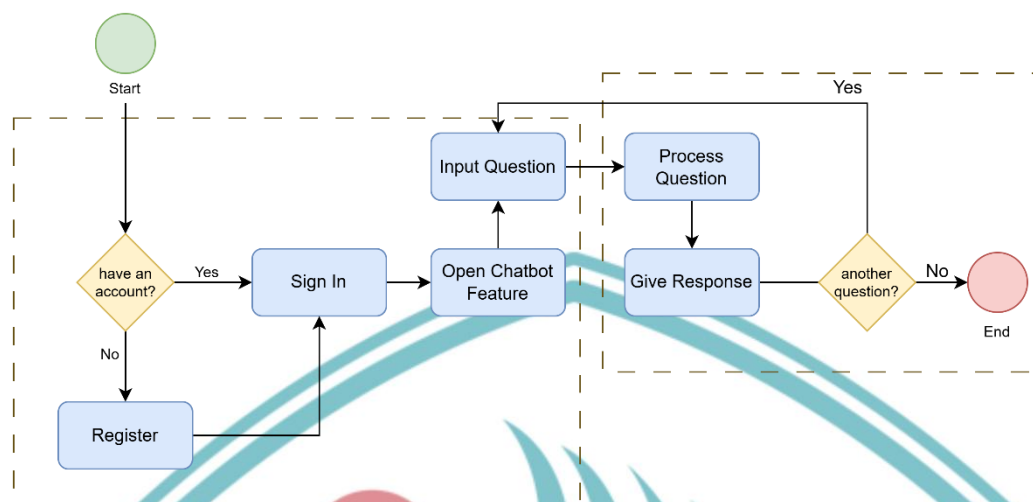
4.2.1 Deskripsi Aplikasi

Sistem *chatbot* yang dikembangkan dalam penelitian ini merupakan aplikasi berbasis web yang dirancang untuk menyediakan layanan informasi terkait skripsi secara otomatis dan interaktif. Aplikasi ini memiliki dua peran utama pengguna, yaitu mahasiswa sebagai pengguna akhir yang mengakses informasi dan admin (staf administrasi atau dosen) yang bertugas mengelola dan memperbarui data. Fungsi utama aplikasi meliputi pendaftaran akun, *login*, pengunggahan dokumen, dan pengajuan pertanyaan terkait proses administrasi. Sistem ini memanfaatkan kemampuan LLM untuk memproses bahasa alami, sehingga dapat memberikan jawaban yang kontekstual dan relevan berdasarkan data yang telah dimasukkan oleh admin. Dengan demikian, aplikasi ini diharapkan dapat mengurangi ketergantungan mahasiswa pada staf administrasi serta meningkatkan efisiensi layanan, sebagaimana tujuan yang telah dirumuskan.



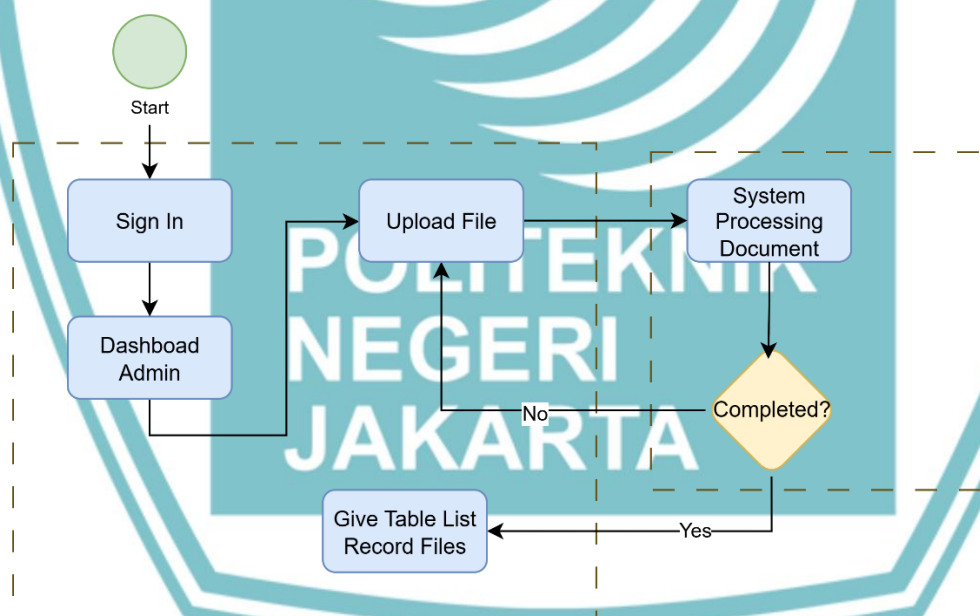
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.1 Alur Diagram Questions and Answer Chatbot Mahasiswa

Pada Gambar 4.1 menggambarkan bagaimana alur dalam bentuk diagram sistem digunakan untuk bertanya hingga sistem memberikan respon jawaban terhadap user.



Gambar 4.2 Alur Diagram Upload Files Pada Admin

Pada Gambar 4.2 menggambarkan bagaimana alur sistem dalam menambah pengetahuan ataupun informasi yang akan sistem gunakan untuk memberikan response sesuai dengan informasi yang didapat.



Hak Cipta :

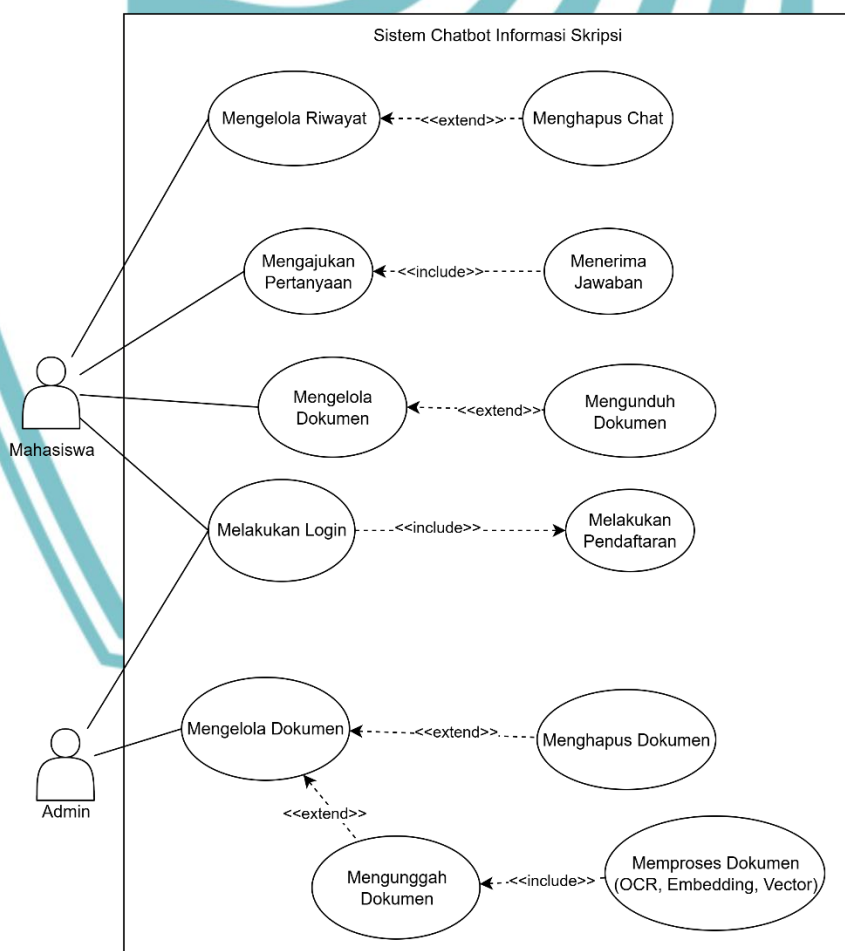
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.2.2 Diagram UML

Perancangan diagram UML (Unified Modelling Language) merupakan salah satu tahap yang penting untuk dilakukan dalam merancang aplikasi. Pada penelitian ini, diagram UML yang digunakan dalam perancangan sistem meliputi *use case diagram* dan *activity diagram*.

4.2.2.1 Use Case Diagram

Use Case Diagram digunakan untuk mengidentifikasi fungsi-fungsi utama sistem dan aktor yang terlibat dalam interaksi dengannya. Dalam konteks penelitian ini, terdapat dua aktor utama: (1) Mahasiswa, yang berperan sebagai pengguna akhir yang membutuhkan informasi skripsi, dan (2) Admin, yang bertanggung jawab atas pengelolaan data dan pembaruan informasi.



Gambar 4.3 Use Case Diagram Sistem



Hak Cipta :

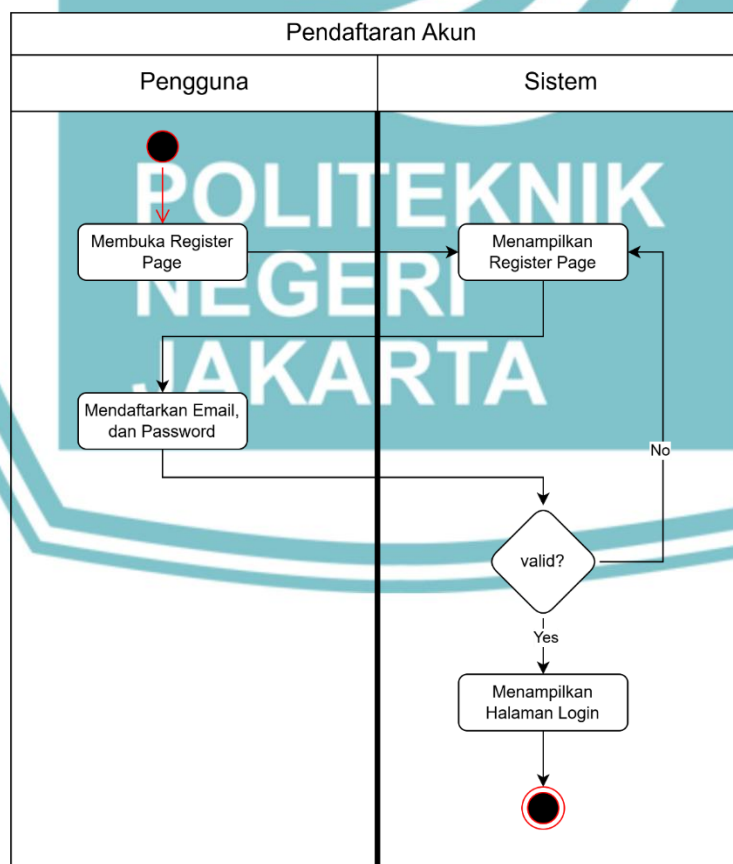
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.2.2.2 Activity Diagram

Activity Diagram digunakan untuk memodelkan alur kerja dari system. Diagram ini menggambarkan langkah-langkah proses secara sekuensial, termasuk keputusan dan percabangan yang mungkin terjadi. Berikut adalah penjelasan untuk masing-masing *activity diagram* yang dirancang:

1) Pendaftaran Akun

Diagram ini mengilustrasikan proses pendaftaran akun oleh mahasiswa. Alur dimulai dari mahasiswa yang mengakses halaman pendaftaran, mengisi data pribadi (seperti *email* dan kata sandi), hingga sistem memverifikasi keunikan data tersebut. Jika data valid, akun dibuat, dan mahasiswa menerima konfirmasi. Proses ini dirancang untuk memastikan aksesibilitas sistem yang mudah bagi pengguna baru, sekaligus menjaga integritas data pengguna.



Gambar 4.4 Activity Diagram Pendaftaran Akun

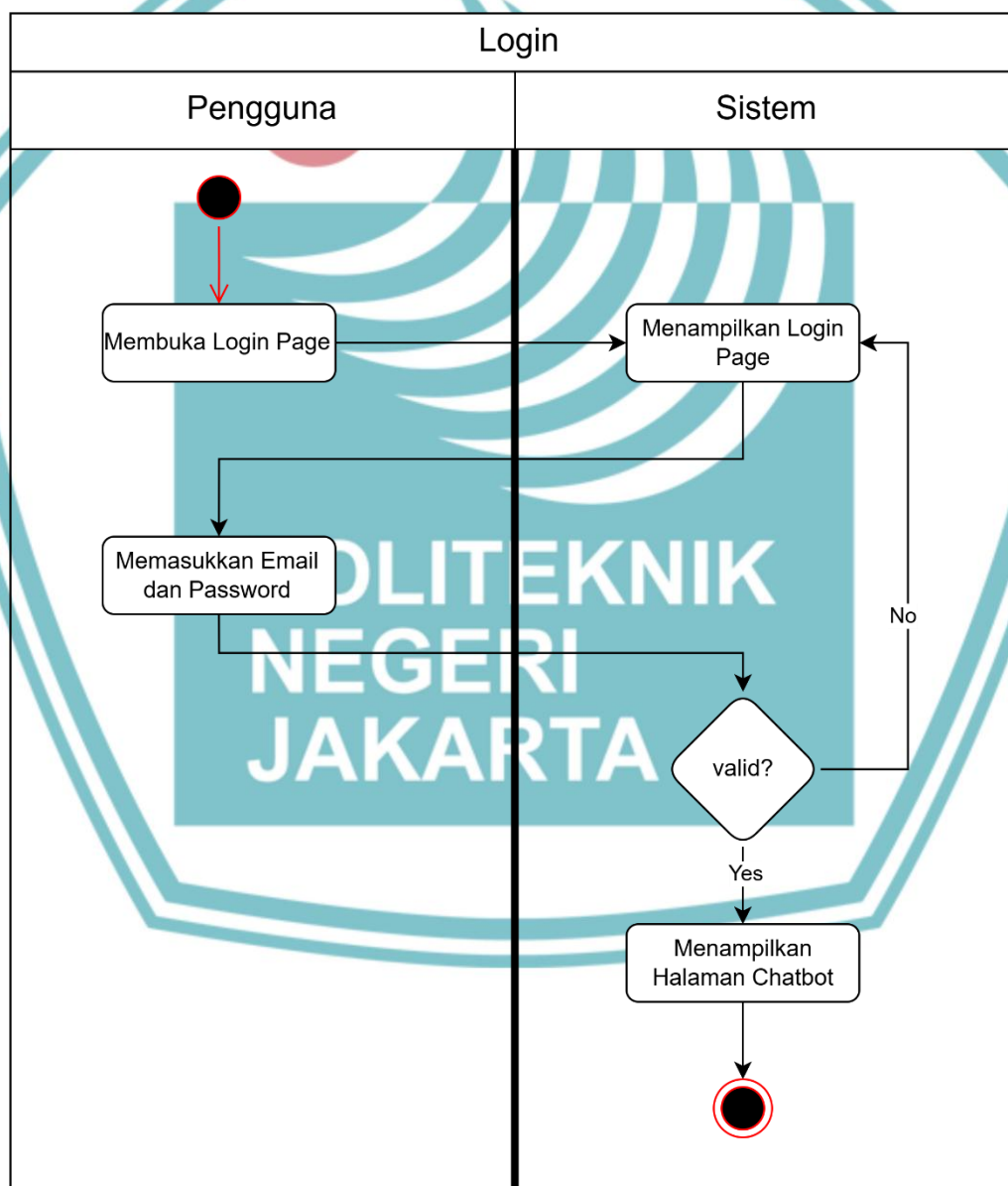


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

2) Login

Activity Diagram untuk *Login* menggambarkan proses autentikasi pengguna. Alur dimulai dengan pengguna memasukkan kredensial (Email dan kata sandi), diikuti oleh verifikasi sistem terhadap *database*. Jika kredensial sesuai, pengguna diarahkan ke halaman utama. Jika gagal, sistem memberikan notifikasi kesalahan. Desain ini mendukung keamanan sistem dan memastikan hanya pengguna terdaftar yang dapat mengakses layanan.



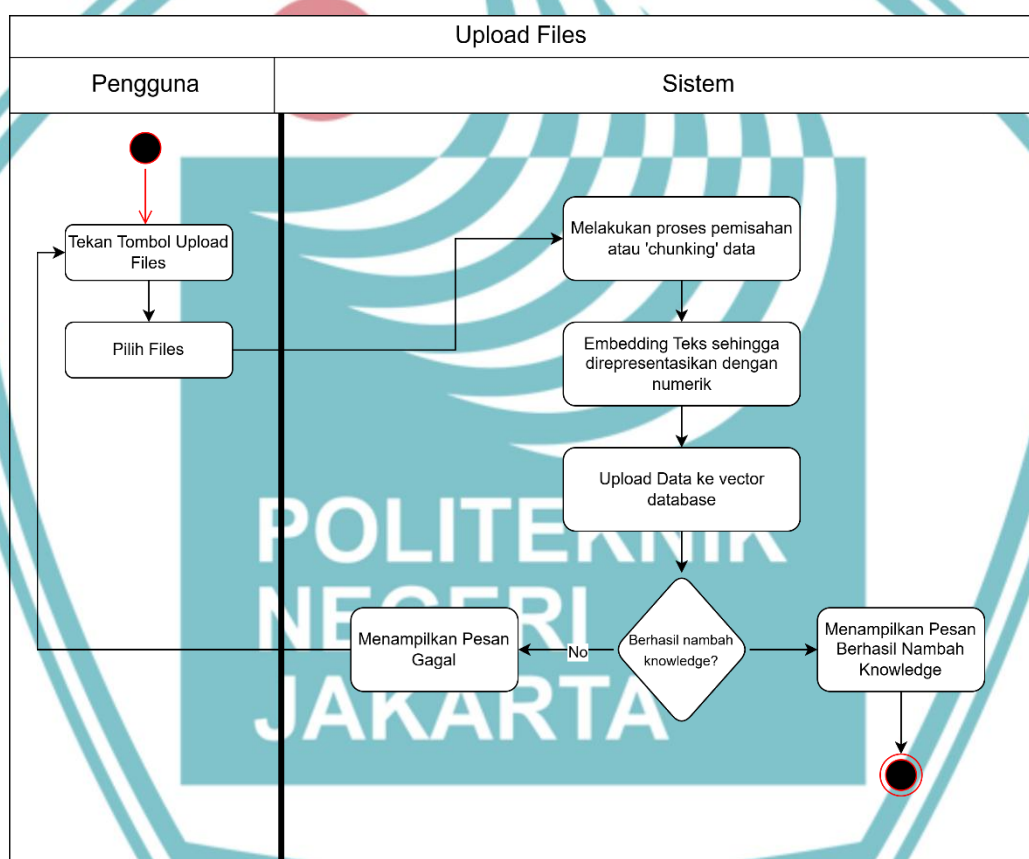
Gambar 4.5 Activity Diagram Login

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3) Upload Files

Diagram ini menunjukkan alur kerja admin dalam mengunggah dokumen atau data terbaru. Proses dimulai dengan admin masuk ke halaman pengelolaan, memilih file (misalnya PDF jadwal skripsi), dan mengunggahnya ke sistem. Sistem kemudian memproses file tersebut untuk diintegrasikan ke dalam basis pengetahuan *chatbot*. Langkah ini krusial untuk memenuhi tujuan pembaruan informasi secara dinamis (lihat Subbab 1.4), sehingga *chatbot* dapat memberikan respon yang selalu relevan dengan data terkini.



Gambar 4.6 Activity Diagram Upload Files

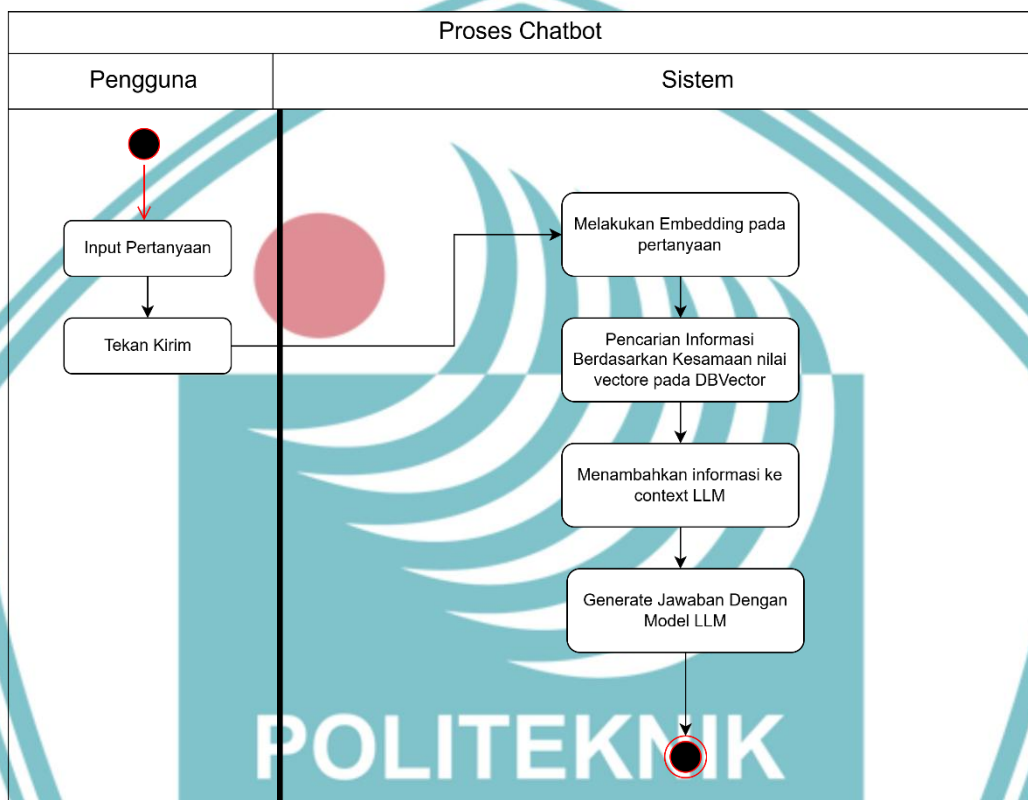
4) Mengajukan Pertanyaan

Activity Diagram ini menggambarkan interaksi mahasiswa dengan *chatbot*. Alur dimulai dari mahasiswa mengajukan pertanyaan melalui antarmuka web, diikuti oleh pemrosesan pertanyaan oleh LLM berdasarkan data yang tersedia. Sistem kemudian menghasilkan jawaban yang dikembalikan kepada

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

mahasiswa. Jika pertanyaan di luar cakupan data, *chatbot* memberikan respons ‘ketidaktahuan’. Proses ini dirancang untuk meningkatkan kecepatan dan akurasi penyediaan informasi, sesuai dengan tujuan bahwa mahasiswa membutuhkan akses informasi yang responsif.



Gambar 4.7 Activity Diagram Interaksi Chat

4.2.3 Arsitektur Sistem

4.2.3.1 Arsitektur Umum

Arsitektur umum sistem merupakan representasi menyeluruh dari hubungan antara komponen *frontend*, *backend*, dan layanan pendukung dalam sistem *chatbot* berbasis *Large Language Model* (LLM) yang dikembangkan. Arsitektur ini dirancang dengan pendekatan berbasis layanan (*service-oriented architecture*) dan modular, guna memastikan skalabilitas, *maintainability*, dan fleksibilitas dalam pengembangan serta integrasi sistem di masa depan.

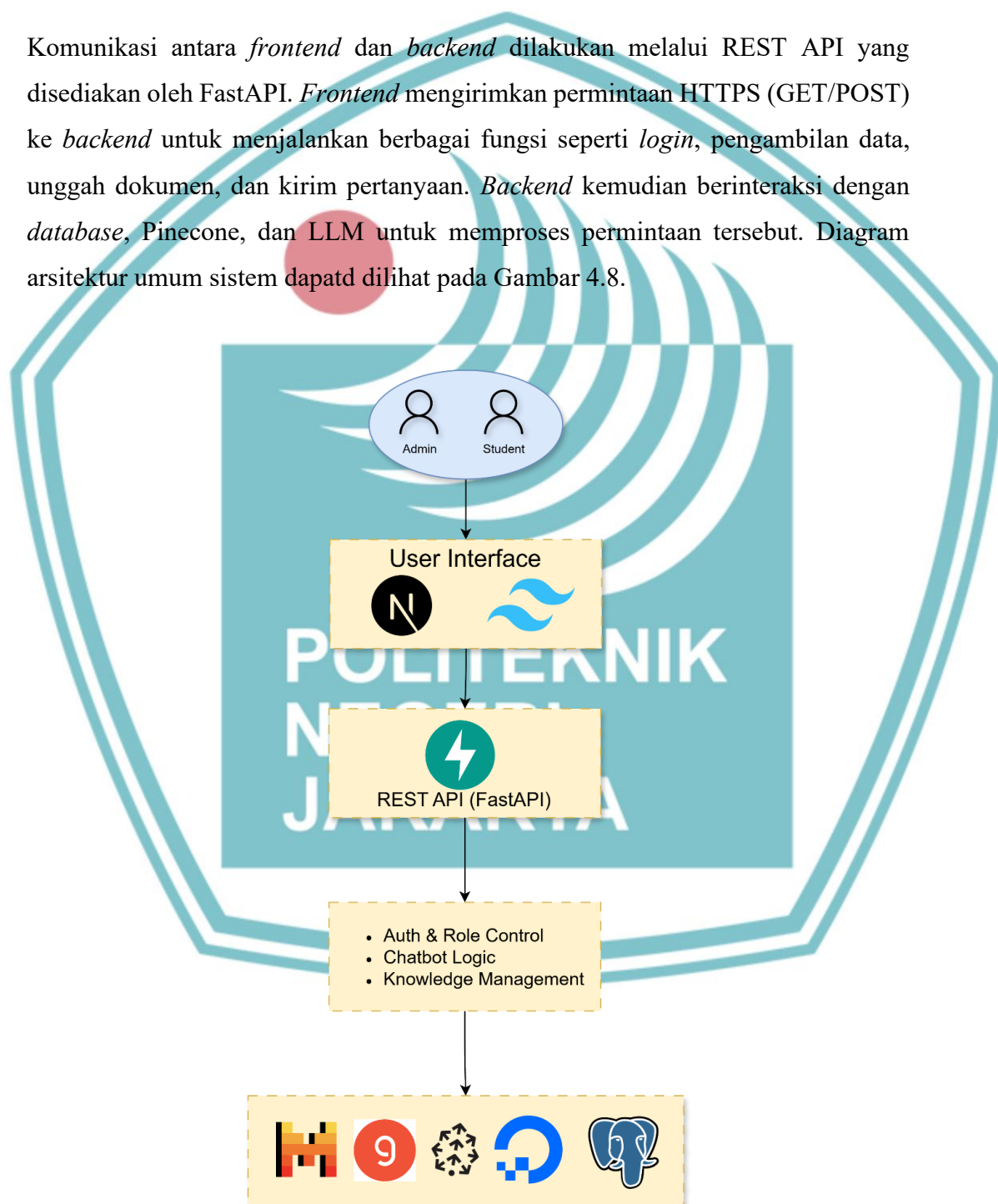


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Sistem ini terdiri atas tiga lapisan utama, yaitu lapisan antarmuka pengguna (*frontend*), lapisan logika aplikasi dan layanan AI (*backend*), serta lapisan penyimpanan data dan *knowledge base* (*storage*). Setiap lapisan berperan penting dalam menjamin fungsionalitas sistem secara keseluruhan.

Komunikasi antara *frontend* dan *backend* dilakukan melalui REST API yang disediakan oleh FastAPI. *Frontend* mengirimkan permintaan HTTPS (GET/POST) ke *backend* untuk menjalankan berbagai fungsi seperti *login*, pengambilan data, unggah dokumen, dan kirim pertanyaan. *Backend* kemudian berinteraksi dengan *database*, Pinecone, dan LLM untuk memproses permintaan tersebut. Diagram arsitektur umum sistem dapat dilihat pada Gambar 4.8.



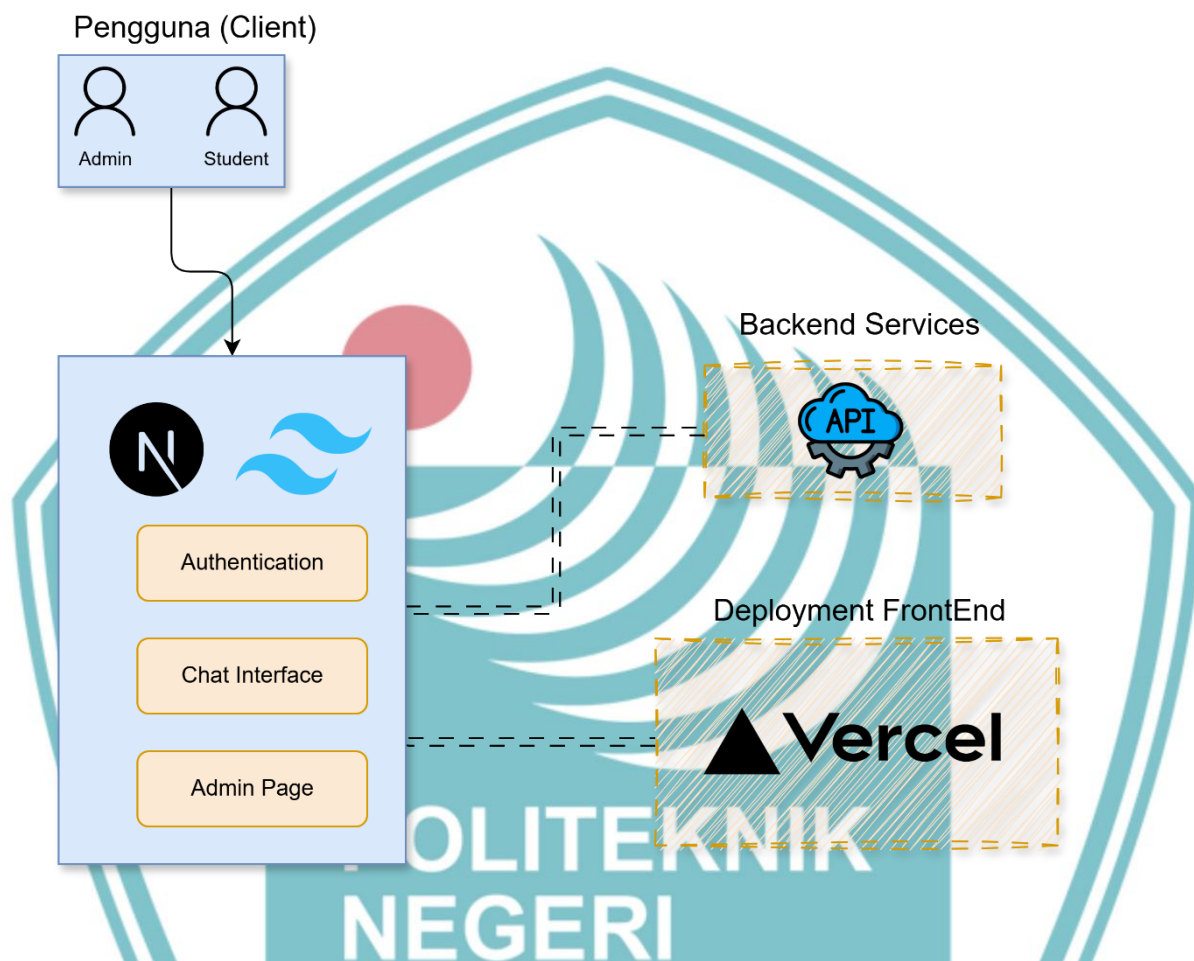
Gambar 4.8 Arsitektur Umum Sistem



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.2.3.2 Arsitektur FrontEnd



Gambar 4.9 Arsitektur Frontend

Arsitektur sistem *frontend* dari aplikasi ini dirancang untuk memberikan antarmuka pengguna yang efisien, responsif dan terintegrasi dengan *backend* AI Chatbot. Aplikasi *frontend* dibangun dengan pendekatan *client-server* menggunakan Next.js sebagai kerangka utama. Aplikasi ini di-*deploy* melalui Vercel dan berkomunikasi dengan *backend* RESTAPI yang mengelola autentikasi, manajemen data dan koneksi ke AI Chatbot. Komponen utama arsitektur *frontend*:

- 1) *User Interface* (UI): Dibangun dengan Next.js dan Tailwind CSS.
- 2) Routing dan Layout: Menggunakan App Router dari Next.js (struktur *folder-based routing*).
- 3) State Management: Menggunakan *hooks* bawaan React dan Context API



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- 4) API Integration Layer: Menangani komunikasi HTTP ke *backend*.
- 5) Role-Based View: Berbasis peran pengguna (mahasiswa dan admin) dengan *protected routes*

4.2.3.3 Arsitektur BackEnd

Sistem ini dibangun dengan arsitektur modular berbasis API menggunakan *framework* FastAPI. Arsitektur sistem secara keseluruhan mengadopsi pendekatan berorientasi layanan (*service-oriented*) dengan pemisahan yang jelas antara komponen-komponen utama.



Gambar 4.10 Arsitektur *Backend*

Sistem menggunakan FastAPI sebagai *framework* utama dengan beberapa *router* yang mengorganisir *endpoint* berdasarkan fungsionalitasnya. Komponen-komponen *backend* disini:

- 1) *Router Layer (Endpoint Handler)* Setiap fungsionalitas (seperti autentikasi, pengelolaan data, dan interaksi dengan *chatbot*) diorganisir dalam modul



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

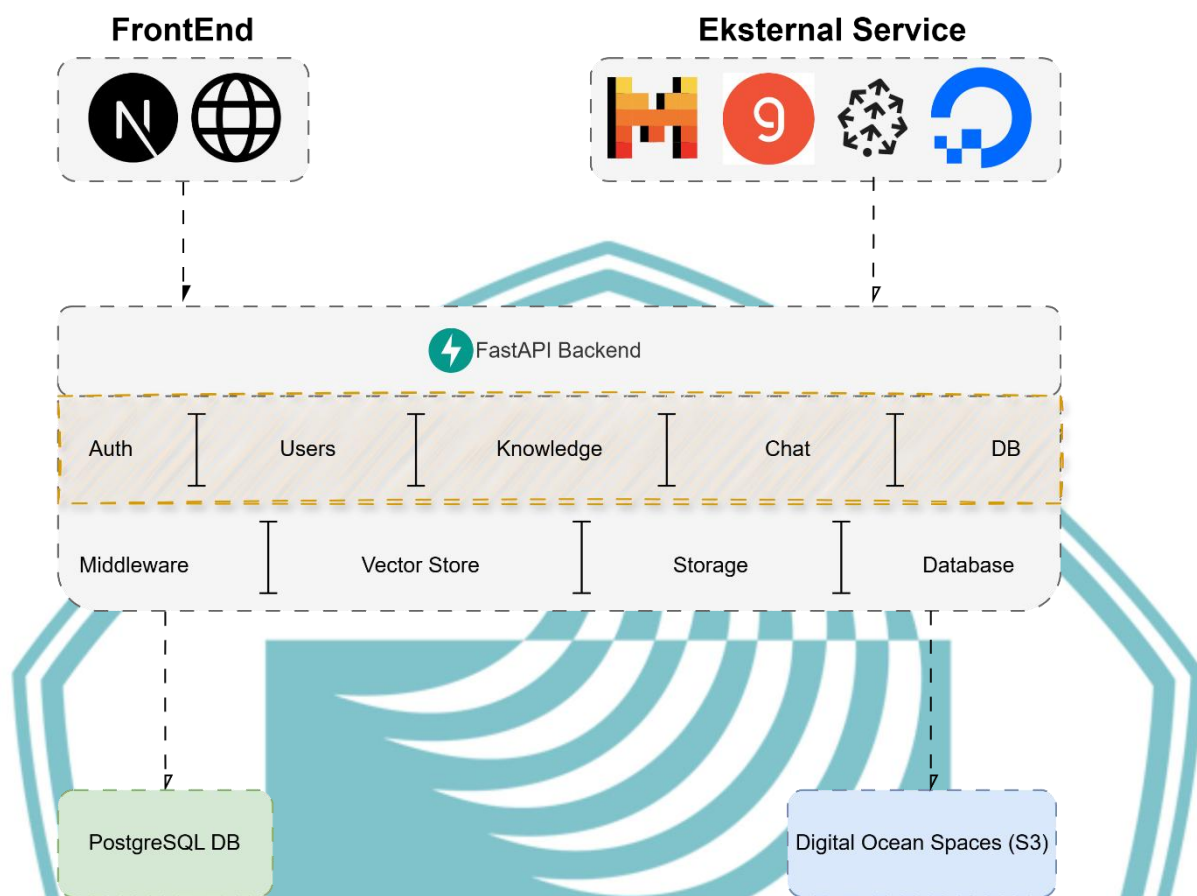
atau *router* terpisah. Hal ini menciptakan struktur kode yang bersih dan mudah dikembangkan.

- 2) Middleware, Middleware digunakan untuk memproses permintaan secara global, seperti: Otentikasi token JWT (JSON Web Token) untuk memverifikasi identitas pengguna pada setiap permintaan. Logging aktivitas pengguna dan CORS middleware untuk mengizinkan permintaan dari domain *frontend*.
- 3) *Services*, Logika inti diletakkan di dalam *service*, yang bertugas mengatur alur kerja setiap permintaan. *Service* ini bertindak sebagai jembatan antara *router* dan data, sehingga memudahkan *debugging* dan *testing* unit secara terpisah.
- 4) Vectore Store, Untuk kebutuhan *Retrieval-Augmented Generation* (RAG), sistem mengintegrasikan:
 - a. Pinecone sebagai *vector* store untuk menyimpan embedding hasil ekstraksi dokumen. Setiap dokumen yang diunggah oleh admin akan diproses menggunakan OCR, diekstrak teksnya, dipecah menjadi chunk, diubah menjadi vektor embedding, lalu disimpan dengan ID unik.
 - b. Model LLM digunakan untuk menjawab pertanyaan berdasarkan konteks yang diambil dari Pinecone. Alur ini menjamin bahwa jawaban yang diberikan bersifat kontekstual dan sesuai dengan isi dokumen yang tersedia.
- 5) *Database*, Sistem menggunakan *database* relasional seperti *PostgreSQL* untuk menyimpan data pengguna, *history chat*, dan metadata dokumen. Interaksi ke *database* dilakukan melalui ORM (*Object Relational Mapping*), yang menyediakan fleksibilitas dalam *query* data dan menjaga integritas relasi antar tabel.
- 6) Virtual Machine, Digunakan agar selalu dapat menangani permintaan dari *frontend* secara on-time 24 jam. *Virtual machine* yang digunakan dari Digital Ocean.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.11 Hubungan Sistem

Arsitektur Sistem ini dengan *FastAPI Backend* yang mana *Auth* untuk mengelola autentikasi pengguna menggunakan JWT (*JSON Web Token*) dan otorisasi berbasis peran (*role-based access control*). Kemudian untuk *User Management* berfungsi untuk mengelola data pengguna, termasuk pendaftaran, pembaruan profil, dan pengambilan data pengguna. *Knowledge Base Management* berfungsi mengelola basis pengetahuan berupa dokumen, gambar, dan audio yang diunggah oleh pengguna. Data ini diproses untuk menghasilkan teks yang dapat digunakan oleh *chatbot*. *Chatbot* berfungsi menyediakan layanan *chatbot* berbasis RAG untuk menjawab pertanyaan pengguna berdasarkan basis pengetahuan.

Pada aplikasi *chatbot* ini, model LLM yang digunakan diambil dari provider yaitu Groq Cloud API yang menyediakan beberapa model LLM secara gratis. Model yang dipilih yaitu *Llama-3.3-70b-versatile*. Sistem menggunakan model Llama 3.3



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

70B yang dihosting oleh Groq, yang merupakan model dengan 70 miliar parameter yang menawarkan keseimbangan optimal antara kualitas respons dan efisiensi. Penting untuk ditekankan bahwa operasional LLM dalam sistem ini tidak berdiri sendiri, melainkan terintegrasi dalam sebuah kerangka kerja *Retrieval-Augmented Generation* (RAG) dengan *Framework* LangChain. Dalam arsitektur RAG, LLM berinteraksi dengan *vector store* (dalam kasus ini, Pinecone) untuk mengambil informasi relevan yang akan digunakan sebagai konteks tambahan sebelum menghasilkan respons terhadap *query* pengguna. Kemampuan Llama 3 cukup akurat memahami dan mensintesis informasi yang diambil dari *knowledge base* melalui Pinecone menjadi faktor penentu kualitas respons *chatbot*. Fitur-fitur arsitektural LLM, seperti panjang konteks yang dapat diproses, efisiensi mekanisme atensi, dan kapasitas model secara keseluruhan, secara langsung memengaruhi seberapa baik LLM dapat memanfaatkan konteks yang disediakan oleh RAG.

Category Benchmark	Llama 3.1 70B	Llama 3.3 70B	Amazon Nova Pro	Llama 3.1 405B	Gemini Pro 1.5	GPT-4o	Claude 3.5 Sonnet
General							
MMLU Chat (0-shot, CoT)	86.0	86.0	85.9	88.6	87.1	87.5	88.9
MMLU PRO (5-shot, CoT)	66.4	68.9	-	73.4	76.1	73.8	77.8
Instruction Following							
IFEval	87.5	92.1	92.1	88.6	81.9	84.6	89.3
Code							
HumanEval (0-shot)	80.5	88.4	89.0	89.0	89.0	86.0	93.7
MBPP EvalPlus (base) (0-shot)	86.0	87.6	-	88.6	87.8	83.9	86.8
Math							
MATH (0-shot, CoT)	67.8	77.0	76.6	73.9	82.9	76.9	78.3
Reasoning							
GPQA Diamond (0-shot, CoT)	48.0	50.5	-	49.0	53.5	47.5	65.0
Tool use							
BFCL v2 (0-shot)	77.5	77.3	-	81.1	80.3	74.0	79.3
Long context							
NIH/Multi-needle	97.5	97.5	-	98.1	94.7	-	99.4
Multilingual							
Multilingual MGSM (0-shot)	86.9	91.1	-	91.6	89.6	90.6	92.8
Pricing*							
1M Input tokens (Cheapest among providers)*	\$0.1	\$0.1	\$0.80	\$1.0	\$1.30	2.5\$	\$3.0
1M Output tokens (Cheapest among providers)*	\$0.4	\$0.4	\$3.20	\$1.8	\$5.0	10.0\$	\$15.0

Gambar 4.12 Benchmark Model Llama 3.

Berdasarkan Gambar 2.1 pemilihan Llama 3.3 70B sudah model Llama 3.3 70B dipilih sebagai model utama untuk penelitian ini dengan pertimbangan sebagai berikut:

- 1) Peningkatan Performa Signifikan dari Generasi Sebelumnya: Llama 3.3 70B menunjukkan peningkatan kapabilitas yang nyata dibandingkan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

dengan pendahulunya, Llama 3.1 70B, pada sejumlah metrik evaluasi penting.

- 2) Kemampuan Multibahasa yang Lebih Baik: Kemampuan multibahasa pada *Multilingual* MGSM juga menunjukkan lompatan signifikan dari 86.9 menjadi 91.1, yang menandakan pemahaman lintas bahasa yang lebih baik pada model dengan parameter 70B
- 3) Kompetitif dengan Model dari Perusahaan Lainnya: model Llama dapat bersaing dengan berbagai model lainnya di era nya (2024).
- 4) Efisiensi Biaya: Salah satu faktor penentu yang paling signifikan adalah efisiensi biaya. Llama 3.3 70B jika melalui Groq Cloud dapat digunakan secara gratis dan sudah berbentuk *endpoint* sehingga tidak perlu lagi menangani bagaimana cara deploy model nya jika download terlebih dahulu. Model Llama 3.3 70B ini adalah model yang paling baik diantara list model yang disediakan oleh Groq.

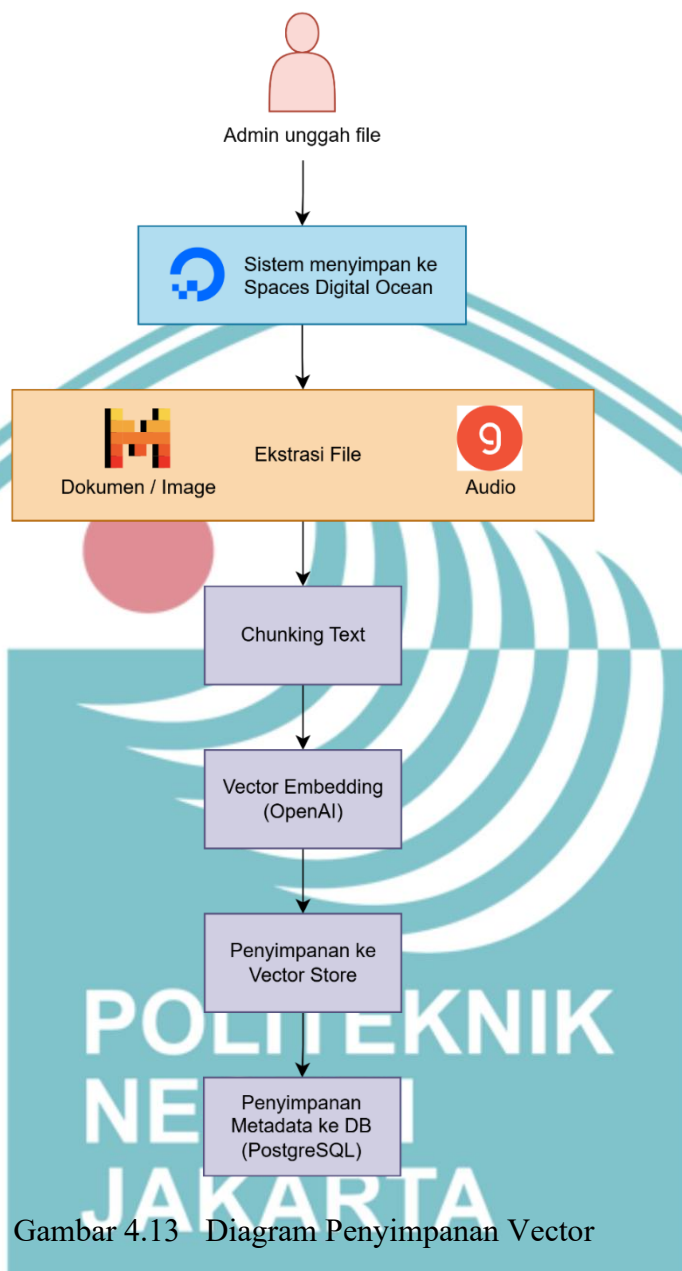
4.2.3.3.1 Sistem Pemrosesan Vector

Pemrosesan dan penyimpanan vektor merupakan tahap fundamental dalam arsitektur *Retrieval-Augmented Generation* (RAG) sistem *chatbot* ini. Tahapan ini bertujuan untuk mengubah korpus pengetahuan yang bersumber dari berbagai format dokumen menjadi representasi numerik (vektor) yang dapat dipahami dan diolah oleh model bahasa. Vektor-vektor ini kemudian disimpan dalam sebuah *database* vektor khusus dalam hal ini menggunakan Pinecone, yang memungkinkan pencarian kemiripan semantik secara efisien untuk mengambil informasi yang relevan terhadap kueri pengguna. Alur kerja pemrosesan dan penyimpanan vektor diimplementasikan dalam modul *knowledge* dan *vector_store* pada sistem *backend*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.13 Diagram Penyimpanan Vector

4.2.3.4 Arsitektur Deployment

Deployment adalah proses menempatkan aplikasi ke lingkungan produksi agar dapat diakses oleh pengguna. Pada proyek *backend* Skripsi *Chatbot*, *deployment* dilakukan dengan memanfaatkan layanan *cloud* DigitalOcean.

a. Infrastruktur Deployment

Aplikasi *backend* di-host pada *server* virtual (*Droplet*) DigitalOcean dengan spesifikasi:

- 1) Sistem Operasi: Ubuntu 22.04 LTS



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- 2) RAM: 2GB
- 3) CPU: 1 Core
- 4) Storage: 50GB SSD
- 5) Lokasi Server: Singapura (SGP1) untuk mengurangi latensi bagi pengguna di Indonesia.

b. Arsitektur Deployment

Arsitektur *deployment* terdiri dari:

- Nginx: Sebagai reverse proxy untuk mengarahkan permintaan HTTP/HTTPS ke aplikasi FastAPI.
- Gunicorn + Uvicorn: Menjalankan aplikasi FastAPI dengan worker untuk performa optimal.
- PostgreSQL: Basis data untuk menyimpan data pengguna, metadata, dan riwayat percakapan.
- Supervisor: Mengelola proses aplikasi agar tetap berjalan oleh Droplet Digital Ocean.
- DigitalOcean Spaces: Penyimpanan file seperti dokumen, gambar, dan audio.

Deployment aplikasi Skripsi *Chatbot* dirancang dengan mempertimbangkan aspek performa, keamanan, skalabilitas, dan efisiensi biaya. Penggunaan layanan cloud DigitalOcean dengan konfigurasi *Nginx*, *Supervisor*, dan *PostgreSQL* memungkinkan aplikasi berjalan dengan andal dan responsif. Integrasi dengan layanan AI eksternal seperti Pinecone, Mistral, dan Groq memberikan fungsionalitas *RAG* yang canggih.

4.3 Implementasi Sistem

Implementasi sistem merupakan pelaksanaan dari perancangan sistem yang mana fase realisasi dari seluruh rancangan sistem yang telah diuraikan pada subbab sebelumnya. Pada tahap ini, arsitektur sistem, perancangan antarmuka pengguna (UI), desain *database*, dan diagram alur proses yang telah didefinisikan, diwujudkan menjadi sebuah sistem aplikasi *chatbot* yang fungsional. Implementasi mencakup pengembangan komponen *frontend*, *backend*, serta konfigurasi layanan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

pendukung untuk memastikan sistem dapat berjalan sesuai dengan kebutuhan fungsional dan non-fungsional yang ditetapkan.

4.3.1 Lingkungan Implementasi

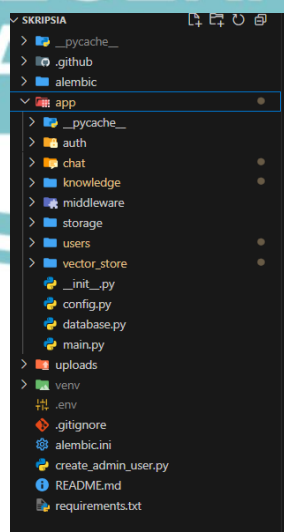
Proses implementasi sistem *chatbot* ini didukung oleh lingkungan pengembangan yang terdiri dari perangkat keras (*Hardware*) dan perangkat lunak (*Software*) spesifik. Pemilihan lingkungan ini didasarkan pada kebutuhan untuk mengakomodasi pengembangan aplikasi berbasis web dengan *framework modern* dan integrasi dengan layanan LLM serta *vector store*.

4.3.2 Implementasi Backend

Implementasi *backend* sistem *chatbot* dikembangkan menggunakan bahasa pemrograman Python dengan *framework* FastAPI dan LangChain, sesuai dengan arsitektur yang dirancang pada Gambar 4.9 dan Gambar 4.10 Fokus utama implementasi *backend* adalah membangun API yang *robust* dan efisien untuk melayani permintaan dari *frontend*, mengelola logika bisnis, berinteraksi dengan *database*, serta mengintegrasikan layanan LLM dan *vector store*.

4.3.2.1 Struktur Proyek dan Modul

Proyek *backend* ini diorganisir secara modular untuk memastikan keterbacaan, pemeliharaan, dan skalabilitas kode.



Gambar 4.14 Struktur Proyek dan Modul Backend



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Struktur direktori utama aplikasi (*app/*) mencakup beberapa modul inti yang masing-masing bertanggung jawab atas fungsionalitas spesifik:

- 1) *auth/*: Modul ini menangani semua aspek terkait autentikasi dan otorisasi pengguna. Ini termasuk proses registrasi pengguna baru, *login* pengguna menggunakan kredensial, pembuatan dan validasi token JWT (JSON Web Token), serta proteksi *endpoint* berdasarkan peran pengguna (*role-based access control*). Implementasi menggunakan *OAuth2PasswordBearer* dari FastAPI dan Pydantic model untuk validasi data.
- 2) *users/*: Bertanggung jawab atas pengelolaan data pengguna, seperti menampilkan profil pengguna yang sedang *login* dan memperbarui informasi profil. Akses ke beberapa fungsi di modul ini mungkin dibatasi hanya untuk pengguna dengan peran admin.
- 3) *knowledge/*: Modul krusial untuk manajemen basis pengetahuan *chatbot*. Ini mencakup fungsionalitas untuk mengunggah berbagai jenis *file* (PDF, gambar, audio), memproses *file* tersebut untuk ekstraksi *teks* (menggunakan OCR dari Mistral untuk dokumen/gambar dan transkripsi audio dari Groq), melakukan *chunking teks*, membuat *embedding*, dan menyimpannya ke dalam *vector store* Pinecone. Modul ini juga menangani pengambilan dan penghapusan *knowledge source*.
- 4) *chat/*: Merupakan inti dari fungsionalitas *chatbot*. Modul ini menangani penerimaan pertanyaan dari pengguna, interaksi dengan layanan LLM (Groq llama-3.3-70b-versatile), pengambilan konteks relevan dari *vector store* Pinecone melalui mekanisme RAG (*Retrieval-Augmented Generation*), dan pengelolaan riwayat percakapan.
- 5) *vector_store/*: Berisi logika untuk berinteraksi dengan Pinecone, termasuk inisialisasi koneksi, penyimpanan *embedding teks*, dan pencarian *similarity* untuk RAG. Menggunakan *OpenAIEmbeddings* untuk menghasilkan vektor dari teks.
- 6) *middleware/*: Direktori ini berisi custom *middleware* yang dapat digunakan, misalnya untuk validasi peran pengguna secara global atau *rate limiting* API.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- 7) *storage/*: Modul ini berisi logika untuk interaksi dengan layanan penyimpanan eksternal yaitu DigitalOcean Spaces, yang digunakan untuk menyimpan *file* basis pengetahuan yang diunggah oleh admin.
- 8) *main.py*: Titik masuk (*entry point*) utama aplikasi FastAPI, di mana semua *router* dari berbagai modul di atas diinisialisasi dan di-include. Di sini juga konfigurasi CORS (*Cross-Origin Resource Sharing*) diatur untuk memungkinkan komunikasi dengan aplikasi *frontend*.
- 9) *config.py*: Mengelola semua konfigurasi aplikasi, termasuk kunci API untuk layanan eksternal, URL *database*, dan pengaturan JWT. Pengaturan dimuat dari *environment variables* menggunakan *Pydantic Settings*.
- 10) *database.py*: Menangani koneksi ke *database* PostgreSQL menggunakan SQLAlchemy ORM dan menyediakan mekanisme untuk sesi *database*.
- 11) *alembic/*: Direktori ini berisi kode dan konfigurasi Alembic untuk melakukan migrasi skema basis data, memastikan bahwa struktur tabel selalu sinkron dengan model data aplikasi.

Struktur ini memungkinkan pengembangan yang terisolasi antar modul dan pengelolaan dependensi. Router FastAPI digunakan untuk mengelompokkan *endpoint* API berdasarkan fungsionalitasnya, seperti yang diilustrasikan pada Gambar 4.13.

4.3.2.2 Endpoint API

Setiap fungsionalitas utama diekspos melalui *endpoint* REST API yang terdefinisi dengan baik, menggunakan metode HTTP yang sesuai (GET, POST, PUT, DELETE).

Tabel 4.4 *Endpoint API*

Autentikasi (/auth)	
POST <i>/auth/register</i>	Menerima data pengguna baru. Melakukan validasi, hashing <i>password</i> , penyimpanan ke <i>database</i> , dan mengembalikan token JWT
POST <i>/auth/token</i>	Menerima kredensial pengguna (<i>username</i> , <i>password</i>). Memverifikasi



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

	kredensial jika berhasil menghasilkan dan mengembalikan token JWT
Pengguna (/users)	
GET <i>/knowledge</i>	Membutuhkan token JWT user admin. Mendapatkan daftar semua <i>knowledge sources</i> yang ada di <i>database</i> .
POST <i>/knowledge/upload</i>	Membutuhkan token JWT dan peran admin. Menerima unggahan files (<i>document</i> , <i>image</i> dan <i>voice</i>). Memproses files (ekstraksi teks, chunking dan embedding), dan menyimpan ke <i>vector store</i> (pinecone) serta metadata ke PostgreSQL. Mengembalikan status sukses atau gagal.
GET <i>/knowledge/file/{knowledge_id}</i>	Membutuhkan token JWT dan akan mendapatkan file <i>knowledge source</i> tertentu dari spaces/storage Digital Ocean untuk didownload
GET <i>/knowledge/preview/{knowledge_id}</i>	Membutuhkan token JWT dan akan mendapatkan preview metadata dari <i>knowledge source</i> tersebut
DELETE <i>/knowledge/{knowledge_id}</i>	Membutuhkan token JWT user admin, dan bertujuan untuk menghapus <i>knowledge source</i> (hanya untuk admin).
Chat (/chat)	
POST <i>/chat/send</i>	Membutuhkan token JWT. Menerima pertanyaan dari pengguna. Memproses pertanyaan melalui alur RAG(<i>query embedding</i> , <i>similarity search</i> di Pinecone, <i>prompt engineering</i> dengan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

	konteks dan interaksi dengan LLM), dan mengembalikan jawaban yang dihasilkan oleh LLM
POST <i>/chat/conversations</i>	Membutuhkan token JWT, Menerima daftar <i>log</i> percakapan oleh user yang sedang <i>login</i> tersebut
POST <i>/chat/conversations/{conversation_id}</i>	Membutuhkan token JWT, Mendapatkan detail dari <i>log conversation</i> tertentu <i>based on id</i>
DELETE <i>/chat/conversations/{conversation_id}</i>	Akan menghapus <i>log percakapan</i> tertentu

4.3.2.3 Implementasi *Middleware*

Implementasi *middleware* pada FastAPI digunakan untuk digunakan untuk menangani tugas-tugas lintas-permintaan (*cross-cutting concerns*), seperti:

Tabel 4.5 Sistem *Middleware*

Middleware	Deskripsi
Otentikasi Token JWT	Pada setiap permintaan ke <i>endpoint</i> yang dilindungi, <i>middleware</i> ini mengekstrak token dari header <i>Authorization</i> . Token kemudian di- <i>decode</i> dan divalidasi (memeriksa <i>signature</i> dan masa berlaku). Jika valid, informasi pengguna diekstrak dari token dan dapat diakses oleh <i>endpoint handler</i> .
CORS (<i>Cross-Origin Resources Sharing</i>)	<i>CORS Middleware</i> dari FastAPI dikonfigurasi untuk mengizinkan permintaan dari domain <i>frontend</i> (Vercel) dan metode HTTPS yang spesifik, serta mengatur <i>header</i> yang diizinkan.
Rate Limiting	Rate limiting diterapkan pada <i>endpoint</i> tertentu menggunakan SlowAPI untuk mencegah penyalahgunaan API.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.3.2.4 Manajemen Konfigurasi

Manajemen konfigurasi aplikasi ditangani secara terpusat dalam modul *app.config.py*. Sistem ini memanfaatkan pustaka *pydantic-settings* untuk memuat konfigurasi dari environment variables, yang merupakan praktik terbaik untuk aplikasi *modern*. Variabel konfigurasi utama meliputi:

- 1) *DATABASE_URL*: URL koneksi ke basis data PostgreSQL.
- 2) *JWT_SECRET_KEY*: Kunci rahasia untuk menandatangani dan memverifikasi token JWT.
- 3) *JWT_ALGORITHM*: Algoritma yang digunakan untuk *JWT-HS256*.
- 4) *ACCESS_TOKEN_EXPIRE_MINUTES*: Durasi validitas token akses.
- 5) Kunci API untuk layanan eksternal: *OPENAI_API_KEY* (untuk *embeddings*), *GROQ_API_KEY* (untuk LLM dan transkripsi audio), *MISTRAL_API_KEY* (untuk OCR), *PINECONE_API_KEY* dan *PINECONE_INDEX_NAME* (untuk *vector store*).
- 6) Pengaturan penyimpanan file: konfigurasi untuk DigitalOcean Spaces (*DO_SPACE_REGION*, *DO_SPACE_ENDPOINT*, *DO_SPACE_KEY*, *DO_SPACE_SECRET*, *DO_SPACE_NAME*, *USE_SPACES*).

Dengan memisahkan konfigurasi dari kode, aplikasi menjadi lebih aman, karena informasi sensitif tidak di-*hardcode*.

4.3.2.5 Logika Inti

Logika inti sistem *backend* dirancang untuk mengorkestrasi alur kerja yang kompleks, mulai dari autentikasi pengguna, manajemen basis pengetahuan, hingga pemrosesan pertanyaan menggunakan *Large Language Model* (LLM) dengan pendekatan *Retrieval-Augmented Generation* (RAG). Bagian ini akan menguraikan alur logika utama yang terjadi di dalam sistem *backend*.

a. Autentikasi dan Otorisasi

Modul ini bertanggung jawab untuk mengelola akses pengguna ke sistem.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

pwd_context = CryptContext(schemes=["bcrypt"], deprecated="auto")

def verify_password(plain_password, hashed_password):
    return pwd_context.verify(plain_password, hashed_password)

def get_password_hash(password):
    return pwd_context.hash(password)

def authenticate_user(db: Session, username: str, password: str):
    user = db.query(User).filter(User.username == username).first()
    if not user:
        return False
    if not verify_password(password, user.password_hash):
        return False
    return user

def create_access_token(data: dict, expires_delta: Optional[timedelta] = None):
    to_encode = data.copy()
    if expires_delta:
        expire = datetime.utcnow() + expires_delta
    else:
        expire = datetime.utcnow() + timedelta(minutes=15)
    to_encode.update({"exp": expire})
    encoded_jwt = jwt.encode(to_encode, settings.JWT_SECRET_KEY, algorithm=settings.JWT_ALGORITHM)
    return encoded_jwt

```

Gambar 4.15 Implementasi Autentikasi

- Registrasi Pengguna: *Endpoint /auth/register* memungkinkan pengguna baru untuk mendaftar dengan menyediakan *username*, *email*, dan *password*. Sebelum membuat pengguna baru, sistem akan memeriksa apakah *username* atau *email* sudah terdaftar. *Password* pengguna di-hash menggunakan *get_password_hash* dari *app.auth.utils.py* (yang menggunakan *passlib.context.CryptContext* dengan skema *bcrypt*) sebelum disimpan ke *database*. Setelah berhasil, token JWT baru akan dibuat dan dikembalikan.
- Login Pengguna: *Endpoint /auth/token* (*app.auth.router.py*) menggunakan *OAuth2PasswordRequestForm* untuk menerima *username* dan *password*. Fungsi *authenticate_user* (*app.auth.utils.py*) memverifikasi kredensial pengguna terhadap data di *database*, dengan membandingkan *hash password*. Jika autentikasi berhasil, token JWT akses dibuat menggunakan *create_access_token*.
- Proteksi Rute: Dependensi seperti *get_current_user* dan *get_current_admin* (*app.auth.dependencies.py*) digunakan untuk melindungi *endpoint* yang memerlukan autentikasi atau hak akses admin. Dependensi ini memvalidasi token JWT yang diterima dalam *header Authorization* dan mengambil data pengguna dari *database*.

b. Interaksi Chatbot Berbasis RAG

Logika *generate response chatbot* diimplementasikan dalam *chat services*, dengan fungsi utama *generate_response*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
def generate_response(query: str, history: List[Tuple[str, str]] = None) -> str:
    """Generate a response using RAG with conversation history"""

    # Initialize LLM
    llm = ChatGroq(model="llama-3.1-70b-versatile")

    # Retrieve relevant context
    context = retrieve_relevant_chunks(query)

    # Build conversation history string
    chat_history = ""
    if history:
        for i, (role, content) in enumerate(history):
            # Skip the most recent user message as it's already in the query
            if i == len(history) - 1 and role == "user":
                continue
            prefix = "User: " if role == "user" else "Assistant: "
            chat_history += f"{prefix}{content}\n\n"

    # Define RAG prompt
    template = """
    You are a helpful assistant specializing in providing information about skripsi (thesis) guidelines and processes.

    Your task is to answer the user's question based on the provided context. If the answer is not in the context or you're unsure, say that you don't know rather than making up information.

    When providing responses:
    - Keep your answers concise and focused on the user's question
    - Format your response in a clear, readable way
    - Maintain a helpful and supportive tone
    - If you give documentation or references, also provide a brief summary of the content

    Chat History:
    {chat_history}

    Context:
    {context}

    User Question: {question}

    Assistant:
    """

    # Build prompt
    prompt = ChatPromptTemplate.from_template(template)

    # Create RAG chain
    rag_chain = (
        {
            "context": lambda x: context,
            "question": lambda x: q,
            "chat_history": lambda x: chat_history
        }
        | prompt
        | llm
        | StrOutputParser()
    )

    # Generate response
    response = rag_chain.invoke(query)

    return response
```

Gambar 4.16 Implementasi Retrieve Context kedalam Model

Fungsi ini menggunakan paradigma RAG (*Retrieval-Augmented Generation*) melalui *LangChain*, yang menggabungkan retrieval dan *generate* untuk respons yang lebih akurat dan kontekstual. *LangChain* sendiri adalah *framework* yang dirancang untuk pengembangan aplikasi berbasis *Large Language Model (LLM)* seperti OpenAI GPT atau model lainnya. *LangChain* memungkinkan pengembang untuk mengintegrasikan komponen NLP secara modular dan efisien (LangChain, 2023).

Pada proses *interaksi chatbot* ini, sebelum generate response melalui LLM akan mengambil konteks yang relevan terlebih dahulu dari *vector database* dalam penelitian ini menggunakan *Pinecone*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
def retrieve_relevant_chunks(query: str, k: int = 5) -> str:
    """Retrieve relevant chunks from vector store based on query"""
    vector_store = get_vector_store()
    docs = vector_store.similarity_search(query, k=k)

    # Format retrieved documents into a single context string
    context = ""
    for i, doc in enumerate(docs):
        context += f"Document {i+1}: \n{doc.page_content} \n\n"

    return context
```

Gambar 4.17 Pengambilan *Chunk* yang Relevan

Ketika sistem memanggil *retrieve_relevant_chunks*, maka terjadi proses untuk mendapatkan potongan dokumen yang paling relevan dengan *query* pengguna dari penyimpanan vektor Pinecone. Metode yang digunakan dalam mencari dokumen yang relevan adalah *similarity_search* function dari Pinecone, dengan mencari nilai-nilai *vector* yang mirip atau mendekati dengan nilai *vector* dari dokumen yang diberikan kemudian mengembalikan *top n chunk* yang memiliki nilai yang paling mendekati. Untuk memperjelas mekanisme kerja *similarity_search* dalam konteks aplikasi *chatbot* ini, perlu dipahami beberapa tahapan inti yang terjadi setelah *query user* diubah menjadi representasi vektor:

- 1) **Penerimaan Vektor Pertanyaan:** Fungsi *similarity_search* pada Pinecone menerima vektor numerik yang merepresentasikan *query* (pertanyaan) pengguna sebagai input utama (Tripathi & pinecone, 2023). Vektor *query* ini dihasilkan oleh model *embedding* yang sama (OpenAIEmbeddings dengan model text-embedding-3-large) yang digunakan pada *knowledge base* yang dijelaskan pada bagian c sub bab ini juga.
- 2) **Nearest Neighbor Search:** Inti dari *similarity_search* adalah melakukan pencarian tetangga terdekat di dalam indeks vektor yang telah dibangun Pinecone. Pinecone memanfaatkan algoritma *Approximate Nearest Neighbor* (ANN) (Pinecone, 2024).
- 3) **Perhitungan Skor Kemiripan:** Selama proses pencarian, Pinecone menghitung skor kemiripan antara vektor *query* dan vektor dalam indeks. Perhitungan ini dilakukan berdasarkan metrik kemiripan yang telah ditentukan ketika indeks Pinecone pertama kali dibuat dan dikonfigurasi. Pada kasus ini menggunakan *Cosine Similarity*, metrik ini mengukur



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

kosinus sudut antara dua vektor A dan B. Nilai Cosine Similarity berkisar antara -1 hingga 1. Dalam konteks *text embeddings*, skor yang lebih tinggi (mendekati 1) menandakan kemiripan semantik yang lebih kuat antara kueri pengguna dan *chunk* dokumen. Pinecone akan menghitung skor ini untuk menemukan vektor-vektor yang paling "dekat" atau memiliki sudut terkecil terhadap vektor *query*.

- 4) Pengambilan *Top-N Chunks*: Setelah skor kemiripan dihitung untuk kandidat-kandidat vektor, Pinecone akan mengurutkan hasilnya berdasarkan skor kemiripan tersebut. *Function similarity_search* kemudian mengembalikan sejumlah N (dalam kasus ini 5) *chunks* dokumen yang memiliki skor kemiripan tertinggi. *Chunks* ini, beserta metadata yang menyertainya (seperti ID sumber dokumen), dianggap sebagai informasi yang paling relevan dari basis pengetahuan untuk menjawab *query* pengguna.
- 5) Konteks untuk LLM: *Chunks* teks yang berhasil diambil melalui *similarity_search* inilah yang kemudian akan disajikan sebagai konteks kepada *Large Language Model* (LLM), yaitu Llama-3.3-70b-versatile. LLM akan menggunakan informasi faktual dari *chunks* ini untuk menghasilkan jawaban yang lebih akurat, relevan, dan berdasar terhadap pertanyaan spesifik pengguna, sesuai dengan prinsip kerja arsitektur *Retrieval-Augmented Generation* (RAG).

Dengan demikian, fungsi *similarity_search* dari Pinecone memainkan peran vital dalam menjembatani pertanyaan pengguna dengan basis pengetahuan yang relevan secara semantik. Kecepatan dan akurasi dari proses pencarian kemiripan ini secara langsung memengaruhi kualitas dan relevansi konteks yang diterima oleh LLM, yang pada gilirannya menentukan kualitas jawaban akhir yang diberikan oleh sistem *chatbot*.

Setelah konteks yang dibutuhkan masuk kedalam LLM maka sebuah prompt yang komprehensif perlu disusun. *Prompt* ini tidak hanya berisi pertanyaan asli dari pengguna, tetapi juga diperkaya dengan *chunks* kontekstual yang telah berhasil diambil dari Pinecone. Selain itu, riwayat percakapan sebelumnya juga dapat



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

diintegrasikan ke dalam prompt untuk menjaga koherensi dan kesinambungan dialog. Dalam implementasi sistem ini, *template prompt* dirancang untuk menginstruksikan LLM agar berperan sebagai asisten yang membantu dalam menyediakan informasi terkait skripsi dan prosesnya, serta menekankan pentingnya menjawab berdasarkan konteks yang diberikan dan menjaga agar jawaban tetap ringkas dan jelas..

```
template = """
You are a helpful assistant specializing in providing information about skripsi (thesis)
guidelines and processes.

Your task is to answer the user's question based on the provided context. If the answer is
not in the context or you're unsure, say that you don't know rather than making up
information.

When providing responses:
- Keep your answers concise and focused on the user's question
- Format your response in a clear, readable way
- Maintain a helpful and supportive tone
- If you give documentation or references, also provide a brief summary of the content

Chat History:
{chat_history}

Context:
{context}

User Question: {question}

Assistant:
"""
```

Gambar 4.18 Template Prompt

Prompt yang telah terstruktur tersebut kemudian dikirim ke model LLM yang dipilih, yaitu Llama-3.3-70b-versatile yang diakses melalui *Groq Cloud API*. LLM memanfaatkan kemampuan pemahaman bahasa alami dan penalaran tingkat lanjut untuk:

- 1) Menganalisis Pertanyaan: Memahami maksud dan kebutuhan informasi spesifik dari pertanyaan pengguna.
- 2) Mensintesis Informasi Kontekstual: Memproses dan menghubungkan informasi yang terdapat dalam *chunks* yang diambil dari Pinecone. LLM tidak hanya menyalin informasi dari konteks, melainkan mampu mensintesis dan merangkum poin-poin penting yang relevan dengan pertanyaan.
- 3) Menghasilkan Jawaban: Berdasarkan pemahamannya terhadap pertanyaan dan sintesis informasi dari konteks, LLM menghasilkan sebuah respons dalam bahasa alami. Arsitektur RAG memastikan bahwa jawaban yang



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

dihasilkan lebih faktual dan berakar pada basis pengetahuan yang tersedia (dokumen-dokumen panduan), sehingga mengurangi potensi kekeliruan pemahaman informasi jika hanya mengandalkan pengetahuan internalnya. Teks jawaban yang telah digenerasi oleh LLM kemudian dikembalikan melalui *backend* ke antarmuka pengguna (aplikasi web) untuk ditampilkan kepada mahasiswa. Jawaban ini diharapkan dapat memberikan informasi yang akurat, relevan, dan mudah dipahami.

c. Pemrosesan dan Penyimpanan Vektor

Proses ini diawali ketika seorang admin mengunggah sumber pengetahuan baru melalui *endpoint /knowledge/upload*. Sistem menerima *file* dalam berbagai format seperti PDF, dokumen Word, gambar, maupun audio. Setelah *file* diterima, sistem melakukan serangkaian langkah pra-pemrosesan dan transformasi sebagai berikut:

1. Ekstraksi Konten Teksual: Langkah pertama adalah mengekstrak konten tekstual mentah dari *file* yang diunggah.
 - a) Untuk *file* dokumen (PDF, DOCX) dan gambar (JPG, PNG), sistem memanfaatkan layanan OCR (Optical Character Recognition) dari Mistral AI (*extract_text_from_document* dan *extract_text_from_image*) layanan ini mampu mengenali dan mengekstrak teks dari konten visual.

```
def extract_text_from_document(file_path: str) -> str:
    """Extract text from PDF or document files using Mistral OCR"""
    try:
        # Check if file exists
        if not os.path.exists(file_path):
            raise FileNotFoundError(f"File not found: {file_path}")

        # Initialize Mistral client
        client = Mistral(api_key=settings.MISTRAL_API_KEY)

        logger.info(f"Uploading document to Mistral OCR: {file_path}")

        # Upload document to Mistral OCR
        uploaded_file = client.files.upload(
            file={
                "file_name": os.path.basename(file_path),
                "content": open(file_path, "rb"),
            },
            purpose="ocr"
        )

        # Retrieve the document URL from the uploaded file
        signed_url_obj = client.files.get_signed_url(file_id=uploaded_file.id)
        signed_url = signed_url_obj.url

        if not signed_url:
            raise ValueError(
                "The retrieved file response does not contain a valid 'document_url'."
            )

        # Process OCR using the document URL
        ocr_response = client.ocr.process(
            model="mistral-ocr-latest",
            document={"type": "document_url", "document_url": signed_url},
            include_image_base64=True,
        )
    except Exception as e:
        logger.error(f"Error in extract_text_from_document: {e}")
        raise
```

Gambar 4.19 Ekstrak Document

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- b) Untuk *file* audio (MP3, WAV, M4A), sistem menggunakan layanan transkripsi audio dari Groq (*extract_text_from_audio*) yang mentransformasi konten suara menjadi teks.

```
def extract_text_from_audio(file_path: str) -> str:
    """Extract text from audio files using Groq API"""
    try:
        import os
        import json
        from groq import Groq

        # Initialize Groq client
        client = Groq()

        logger.info(f"Transcribing audio file: {file_path}")

        # Open the audio file
        with open(file_path, "rb") as audio_file:
            # Create a transcription of the audio file
            transcription = client.audio.transcriptions.create(
                file=audio_file, # Required audio file
                model="whisper-large-v3-turbo", # Required model to use for transcription
                prompt="Specify context or spelling",
                response_format="verbose_json",
                timestamp_granularities=["word", "segment"],
                language="id",
                temperature=0.0
            )
```

Gambar 4.20 Ekstrak Audio

2. Segmentasi Teks (*Text Chunking*): Teks yang telah diekstrak seringkali terlalu panjang untuk diproses secara efektif oleh model *embedding* atau untuk dijadikan konteks tunggal bagi LLM.

```
# Split text into chunks
text_splitter = RecursiveCharacterTextSplitter(
    chunk_size=1000,
    chunk_overlap=200,
    separators=["\n\n", "\n", ". ", " ", ""]
)
```

Gambar 4.21 Segmentasi Teks

Oleh karena itu, teks tersebut dipecah menjadi segmen-segmen yang lebih kecil atau *chunks*. Sistem menggunakan *RecursiveCharacterTextSplitter* dari pustaka *langchain_text_splitters*.

3. Generasi Vektor *Embedding*: Setiap *chunk* teks yang dihasilkan kemudian ditransformasikan menjadi representasi vektor numerik padat, yang dikenal sebagai *embedding*. *Embedding* ini mengambil makna semantik dari teks, di mana *chunks* dengan makna serupa akan memiliki representasi vektor yang berdekatan dalam ruang vektor. Sistem menggunakan model



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

embedding text-embedding-3-large dari OpenAI, yang diakses melalui *OpenAIEmbeddings* dari pustaka *langchain_openai*.

4. Penyimpanan Vektor dan Metadata: Vektor *embedding* yang telah digenerasi, beserta dengan *metadata* yang relevan, disimpan ke dalam *database* vektor Pinecone. Fungsi *store_chunks_in_pinecone* bertanggung jawab atas penyimpanan ini. Metadata yang disimpan untuk setiap vektor *chunk* meliputi:

- *source*: Nama *file* asli dari mana teks tersebut berasal.
- *source_id*: ID unik dari *knowledge source* di basis data PostgreSQL, yang mengaitkan *chunk* dengan dokumen induknya.
- *chunk*: Nomor indeks dari *chunk* tersebut dalam dokumen aslinya. Setiap *chunk* juga diberikan ID unik dalam Pinecone untuk memudahkan pengambilan atau penghapusan spesifik jika diperlukan.

Setelah semua *chunk* dari sebuah *file* berhasil diproses dan disimpan sebagai vektor, status dari *KnowledgeSource* dalam *database* PostgreSQL diperbarui menjadi "*completed*". Jika terjadi kesalahan selama proses, status akan diperbarui menjadi "*failed*".

Alur pemrosesan dan penyimpanan vektor ini memastikan bahwa seluruh pengetahuan sistem selalu dalam format yang siap untuk mendukung mekanisme RAG. Ketika pengguna mengajukan pertanyaan, pertanyaan tersebut juga akan diubah menjadi vektor *embedding* menggunakan model yang sama (*text-embedding-3-large*). Kemudian, sistem melakukan pencarian kemiripan (*similarity search*) pada Pinecone untuk menemukan *chunks* vektor yang paling relevan (paling dekat secara semantik) dengan vektor pertanyaan pengguna, sebagaimana diimplementasikan dalam fungsi *retrieve_relevant_chunks*. *Chunks* inilah yang kemudian digunakan sebagai konteks tambahan bagi LLM untuk menghasilkan jawaban yang akurat dan informatif. Dengan demikian, kualitas pemrosesan dan efisiensi penyimpanan vektor menjadi penentu utama kemampuan sistem *chatbot* dalam memahami dan merespons pertanyaan pengguna secara cerdas dan kontekstual berdasarkan korpus pengetahuan yang tersedia.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.3.2.6 Deployment

Deployment sistem merupakan proses pengunggahan dan pengaturan aplikasi agar dapat berjalan pada *server* produksi. Pada penelitian ini, aplikasi *backend* dikembangkan menggunakan *framework* FastAPI dan dideploy pada layanan DigitalOcean Droplet.

```
cloud.digitalocean.com/droplets/495906355/terminal/ui/?os_user=root
GNU nano 6.2 /etc/nginx/sites-available/skripsia
server {
    server_name vidimarpaung.tech www.vidimarpaung.tech;

    location / {
        proxy_pass http://127.0.0.1:8000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }

    client_max_body_size 50M;

    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/letsencrypt/live/vidimarpaung.tech/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/vidimarpaung.tech/privkey.pem; # managed by Certbot
    include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot
    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot
}

server {
    if ($host = vidimarpaung.tech) {
        return 301 https://$host$request_uri;
    } # managed by Certbot

    listen 80;
    server_name vidimarpaung.tech www.vidimarpaung.tech;
    return 404; # managed by Certbot
}
```

Gambar 4.22 Konfigurasi Nginx

Nginx digunakan sebagai reverse proxy untuk mengarahkan permintaan HTTP ke aplikasi *backend*. Konfigurasi Nginx disimpan dalam file `/etc/nginx/sites-available/skripsia`. Proses *deployment* aplikasi *backend* berhasil dilakukan menggunakan DigitalOcean Droplet dengan konfigurasi yang mendukung kebutuhan sistem. Aplikasi dapat diakses melalui domain untuk komunikasi data.

4.3.3 Implementasi *Frontend*

Implementasi *frontend* sistem *chatbot* menggunakan arsitektur *modern* berbasis Next.js 15.3.1 dengan React 19.0.0 sebagai *library* utama untuk membangun antarmuka pengguna. Pemilihan teknologi ini didasarkan pada beberapa pertimbangan strategis untuk mendukung pengembangan aplikasi *web* yang responsif, performa tinggi, dan mudah di-*maintain*.

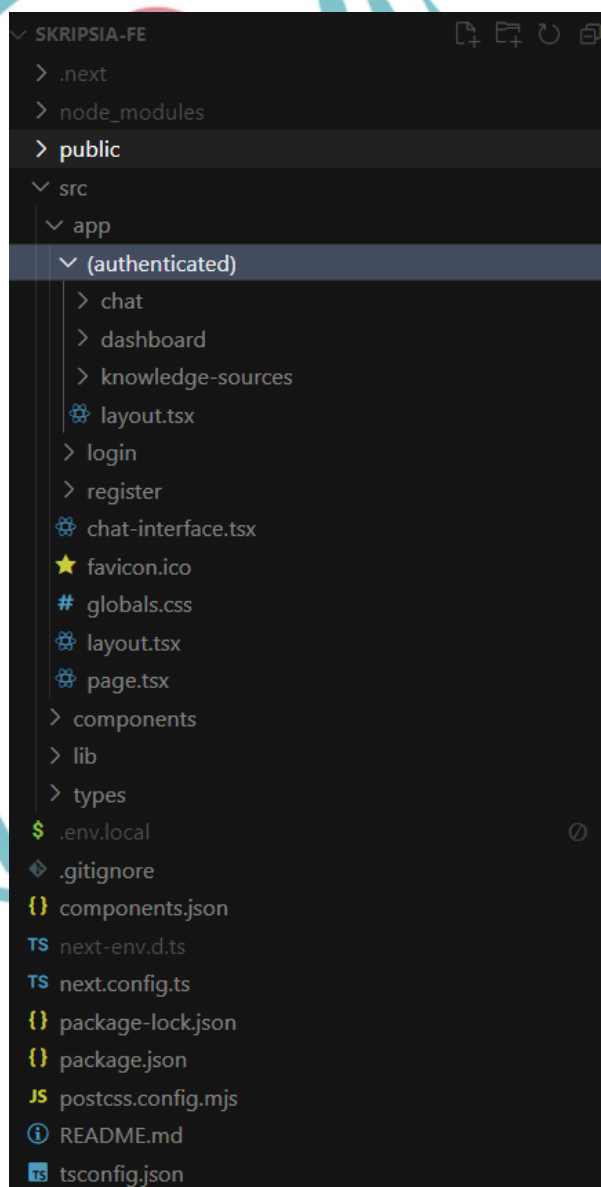


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.3.3.1 Struktur Proyek dan Komponen Utama

Struktur proyek *frontend* diorganisasi secara modular untuk mendukung skalabilitas dan kemudahan pemeliharaan. Direktori utama `src/app` mengikuti konvensi Next.js App Router, yang memungkinkan *routing* berbasis direktori dan pengelolaan *layout* secara hierarkis. Komponen-komponen React yang dapat digunakan kembali ditempatkan dalam direktori `src/components`, yang dikelompokkan lebih lanjut berdasarkan fungsionalitasnya seperti *auth*, *admin*, *chat*, dan *ui*.



Gambar 4.23 Struktur Folder



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Beberapa komponen utama yang diimplementasikan meliputi:

- 1) Tata Letak Aplikasi: Merupakan komponen dasar yang membungkus seluruh halaman aplikasi. *RootLayout (src/app/layout.tsx)* bertanggung jawab untuk menginisialisasi penyedia tema (*ThemeProvider*) dan penyedia otentikasi (*AuthProvider*) yang membungkus seluruh aplikasi, memastikan bahwa status otentikasi dan preferensi tema dapat diakses secara global. Sementara itu, *AuthenticatedLayout* secara spesifik menangani tata letak untuk halaman-halaman yang memerlukan otentikasi pengguna, termasuk menampilkan komponen navigasi utama (*Navigation*) dan memastikan pengguna dialihkan ke halaman *login* jika belum terotentikasi.
- 2) Navigasi: Komponen ini menyediakan antarmuka navigasi utama bagi pengguna yang sudah terotentikasi. Navigasi ini bersifat dinamis, menampilkan tautan yang relevan berdasarkan peran pengguna (admin atau mahasiswa), seperti akses ke *dashboard* admin, halaman *chat*, riwayat percakapan, dan daftar sumber pengetahuan. Komponen ini juga mengintegrasikan fungsionalitas *logout* dan *toggle* tema, serta dirancang responsif untuk perangkat *mobile* dengan menu *dropdown*.
- 3) Manajemen Tema: Sistem mendukung tema terang (*light*) dan gelap (*dark*), serta kemampuan untuk mengikuti preferensi sistem pengguna. *ThemeProvider* menggunakan *React Context* API untuk menyediakan status tema saat ini dan fungsi untuk mengubahnya. Perubahan tema disimpan dalam *localStorage* untuk persistensi. Komponen *ThemeToggle* menyediakan antarmuka visual bagi pengguna untuk beralih antar tema..

4.3.3.3 Implementasi Autentikasi dan Manajemen Sesi

Implementasi autentikasi pengguna merupakan aspek fundamental untuk mengamankan akses ke fitur-fitur sistem dan personalisasi pengalaman pengguna. Sistem ini menggunakan pendekatan *token-based authentication* dengan JSON Web Token (JWT).

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
const AuthContext = createContext<AuthContextType | undefined>(undefined)

export function AuthProvider({ children }: { children: ReactNode }) {
  const [user, setUser] = useState<User | null>(null)
  const [isLoading, setIsLoading] = useState(true)
  const [error, setError] = useState<string | null>(null)

  useEffect(() => {
    const checkAuth = async () => {
      const token = localStorage.getItem('token')
      if (!token) {
        setIsLoading(false)
        return
      }

      try {
        const response = await fetch(`${API_BASE_URL}/users/me`, {
          headers: {
            Authorization: `Bearer ${token}`,
          },
        })

        if (response.ok) {
          const userData = await response.json()
          setUser(userData)
        } else {
          localStorage.removeItem('token')
        }
      } catch (err) {
        console.error('Authentication error:', err)
        localStorage.removeItem('token')
      } finally {
        setIsLoading(false)
      }
    }
  })
}
```

Gambar 4.24 Implementasi AuthProvider

- 1) *AuthProvider*: Komponen *AuthProvider* memanfaatkan React Context API untuk menyediakan status otentikasi pengguna (seperti objek user yang berisi ID, nama pengguna, email, dan peran) dan fungsi-fungsi terkait otentikasi (*login*, *register*, *logout*) secara global ke seluruh komponen aplikasi. Saat aplikasi dimuat, *AuthProvider* memeriksa keberadaan token JWT di *localStorage*. Jika token ditemukan, ia akan mencoba memvalidasi token tersebut dengan mengirim permintaan ke *endpoint* */users/me* pada *backend* untuk mengambil data pengguna.
- 2) *Formulir Login dan Registrasi*: Komponen *LoginForm* menangani proses masuk pengguna dengan mengirimkan nama pengguna dan kata sandi ke *endpoint* */auth/token*. Setelah berhasil, token JWT yang diterima dari *backend* disimpan dalam *localStorage*, dan status pengguna diperbarui dalam *AuthContext*. *RegisterForm* memungkinkan pengguna baru untuk membuat akun dengan mengirimkan nama pengguna, email, kata sandi, dan



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

peran ke *endpoint* /auth/register. Validasi dasar seperti konfirmasi kata sandi juga dilakukan di sisi klien.

- 3) *Protected Route* dan Kontrol Akses Berbasis Peran: Sistem *routing* menggunakan Next.js App Router dengan implementasi *protected routes* melalui direktori (*authenticated*). Komponen *AuthenticatedLayout* secara otomatis memeriksa status otentikasi pengguna. Jika pengguna belum terotentikasi dan mencoba mengakses rute dalam direktori ini (kecuali halaman *login/register*), mereka akan dialihkan ke halaman *login*. Lebih lanjut, halaman *dashboard* utama melakukan pengalihan berdasarkan peran pengguna (*user.role*). Pengguna dengan peran 'admin' dialihkan ke */dashboard/admin*, sementara pengguna 'student' dialihkan ke */chat*. Hal ini memastikan bahwa pengguna hanya dapat mengakses fitur yang sesuai dengan hak akses mereka.
- 4) *Error Handling*: Penanganan kesalahan (*error handling*) yang komprehensif diimplementasikan untuk berbagai skenario kegagalan, seperti kesalahan jaringan atau kredensial yang tidak valid saat *login*. Pesan kesalahan yang informatif ditampilkan kepada pengguna untuk memberikan umpan balik yang jelas.

4.3.3.4 Implementasi Antarmuka Interaksi *Chatbot*

Komponen *ChatInterface* merupakan inti dari sistem yang mengimplementasikan antarmuka percakapan dengan AI. Komponen ini dirancang untuk menyediakan pengalaman pengguna yang intuitif dan interaktif dalam berkomunikasi dengan *chatbot*.

```
export async function sendMessage(message: string, conversation_id?: number): Promise<{message: Message, conversation_id: number}> {
  const headers = {
    'Content-Type': 'application/json',
    ...getAuthHeader() as Record<string, string>
  }

  const response = await fetch(`${API_BASE_URL}/chat/send`, {
    method: 'POST',
    headers,
    body: JSON.stringify({
      message,
      conversation_id
    })
  })

  return handleApiResponse(response)
}
```

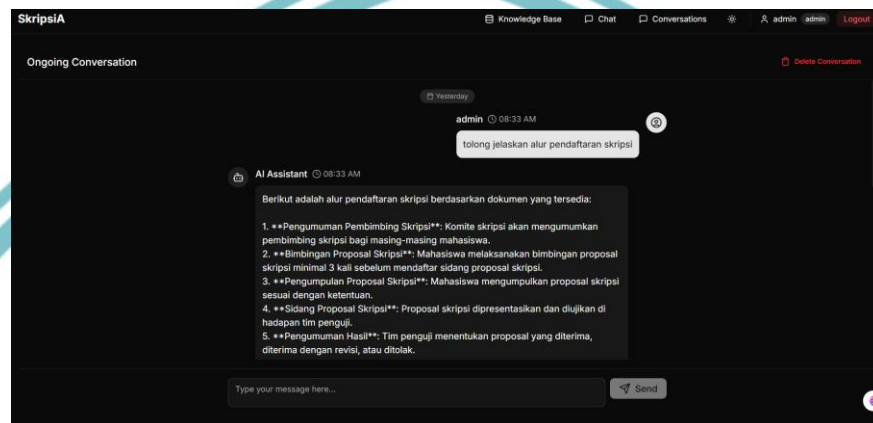
Gambar 4.25 Implementasi Send Message Request

Fitur-fitur utama yang diimplementasikan pada antarmuka *chat* meliputi:

Hak Cipta :

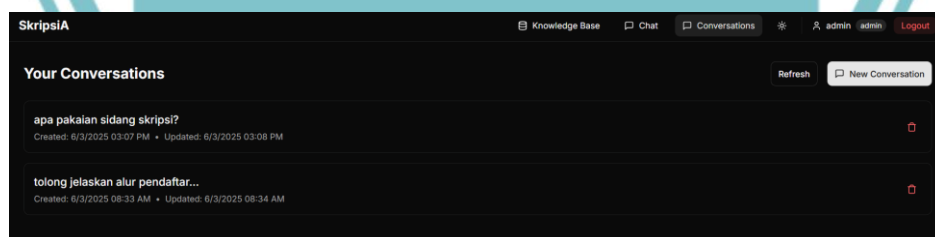
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- 1) Pengiriman dan Penerimaan Pesan Secara Asinkron (*Real-time Messaging*): Pengguna dapat mengetikkan pertanyaan pada komponen Textarea dan mengirimkannya dengan menekan tombol "*Send*" atau tombol Enter. Pesan pengguna segera ditampilkan di antarmuka, diikuti dengan indikator pemrosesan saat sistem menunggu respons dari *backend*. Setelah respons diterima, pesan dari *chatbot* juga ditampilkan secara dinamis.



Gambar 4.26 Tampilan Interaksi Chatbot

- 2) Manajemen Percakapan (*Conversation Management*): Sistem mendukung pengelolaan percakapan. Pengguna dapat memulai percakapan baru atau melanjutkan percakapan yang sudah ada jika *conversation_id* diberikan. Terdapat pula fungsionalitas untuk menghapus percakapan yang sedang berlangsung. Daftar percakapan yang pernah dilakukan pengguna dapat diakses melalui halaman terpisah (*src/app/(authenticated)/chat/conversations/page.tsx*), yang memungkinkan pengguna untuk melihat kembali atau melanjutkan interaksi sebelumnya.



Gambar 4.27 Manajemen Percakapan

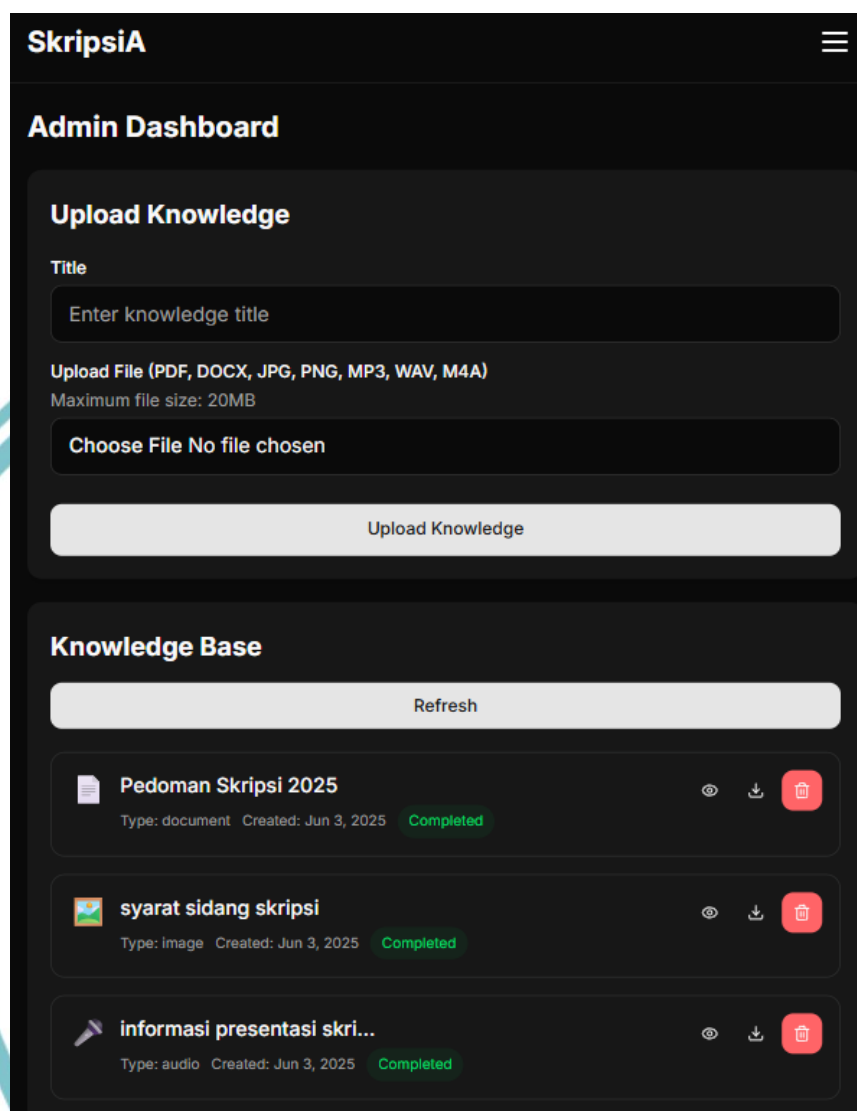
- 3) Atribusi Sumber (*Source Attribution*): Untuk meningkatkan transparansi dan kepercayaan, respons dari AI dapat menyertakan informasi mengenai sumber pengetahuan yang digunakan untuk menghasilkan jawaban tersebut.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Sumber-sumber ini ditampilkan bersama dengan tingkat relevansinya, dan pengguna dapat melihat detail sumber tersebut.



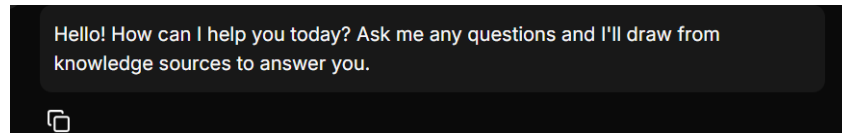
Gambar 4.28 Sources Attribution

- 4) Fitur Tambahan: Antarmuka *chat* juga dilengkapi dengan fitur untuk menyalin respons AI ke *clipboard* (*Copy to Clipboard*), yang memudahkan pengguna untuk menggunakan kembali informasi yang diberikan. Indikator visual (*loading states*) juga ditampilkan saat AI sedang memproses permintaan, memberikan umpan balik yang jelas kepada pengguna mengenai status sistem. Tampilan pesan dibedakan berdasarkan pengirim (pengguna atau AI) menggunakan ikon dan gaya visual yang berbeda untuk mempermudah identifikasi.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.29 Fitur copy to clipboard

4.3.3.5 Implementasi Fitur Administratif

Untuk admin, *frontend* menyediakan antarmuka khusus untuk mengelola basis pengetahuan sistem.

- 1) Pengunggahan Sumber Pengetahuan: Admin dapat mengunggah berbagai jenis *file* (PDF, DOCX, gambar, audio) sebagai sumber pengetahuan baru. Formulir pengunggahan mencakup kolom judul dan pemilihan *file*. Terdapat validasi ukuran *file* di sisi klien (maksimum 20MB) untuk mencegah pengunggahan *file* yang terlalu besar. Setelah pengunggahan berhasil, sistem memberikan notifikasi sukses beserta status pemrosesan dokumen.

```
const handleFileChange = (e: React.ChangeEvent<HTMLInputElement>) => {
  if (e.target.files && e.target.files[0]) {
    const selectedFile = e.target.files[0]
    if (selectedFile.size > MAX_FILE_SIZE_MB * 1024 * 1024) {
      setError(`File size should not exceed ${MAX_FILE_SIZE_MB}MB`)
      setFile(null)
      return
    }
    setFile(selectedFile)

    // Auto-detect file type
    const fileExtension = selectedFile.name.split('.').pop()?.toLowerCase() || ''
    if (['pdf', 'docx', 'doc', 'txt'].includes(fileExtension)) {
      setFileType('document')
    } else if (['jpg', 'jpeg', 'png', 'gif'].includes(fileExtension)) {
      setFileType('image')
    } else if (['mp3', 'wav', 'm4a'].includes(fileExtension)) {
      setFileType('audio')
    }
  }
}
```

Gambar 4.30 Pengunggahan Sources

- 2) Daftar dan Manajemen Sumber Pengetahuan: Halaman ini menampilkan daftar semua sumber pengetahuan yang telah diunggah. Setiap item dalam daftar menunjukkan judul, tipe *file* (dengan ikon yang sesuai), tanggal pembuatan, dan status pemrosesan (*processing*, *completed*, *failed*). Admin dapat melakukan pratinjau (*preview*), mengunduh (*download*), atau



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

menghapus sumber pengetahuan. Fitur pratinjau membuka konten dalam tab baru, dengan tampilan yang disesuaikan berdasarkan tipe *file* (misalnya, PDF *viewer*, pemutar audio, atau tampilan gambar). Untuk tindakan penghapusan, digunakan komponen dialog konfirmasi (`src/components/ui/confirmation-dialog.tsx`) untuk mencegah penghapusan yang tidak disengaja. Daftar ini juga dilengkapi dengan paginasi untuk menangani jumlah data yang besar.

4.3.3.6 Implementasi Fitur Mahasiswa

Selain antarmuka *chat*, mahasiswa juga memiliki akses ke daftar sumber pengetahuan.

- 1) Daftar Sumber Pengetahuan Mahasiswa: Mirip dengan daftar admin, halaman ini (`src/app/(authenticated)/knowledge-sources/page.tsx`) menampilkan sumber-sumber pengetahuan yang tersedia bagi mahasiswa. Namun, fungsionalitas yang tersedia terbatas pada pratinjau dan pengunduhan, tanpa opsi untuk menghapus atau mengelola sumber tersebut. Hal ini bertujuan agar mahasiswa dapat melihat materi rujukan yang digunakan oleh *chatbot*.

```
export function KnowledgeSourceList() {
  const [knowledgeItems, setKnowledgeItems] = useState<KnowledgeItem[]>([])
  const [total, setTotal] = useState(0)
  const [isLoading, setIsLoading] = useState(true)
  const [error, setError] = useState<string | null>(null)
  const [page, setPage] = useState(0)
  const limit = 10
  const [isPreviewLoading, setIsPreviewLoading] = useState<number | null>(null)

  const fetchKnowledgeItems = async () => {
    setIsLoading(true)
    setError(null)

    try {
      const headers = getAuthHeader() as Record<string, string>

      const response = await fetch(`${API_BASE_URL}/knowledge/?skip=${page * limit}&limit=${limit}`, {
        headers,
      })

      if (!response.ok) {
        throw new Error('Failed to fetch knowledge items')
      }

      const data = await response.json() as KnowledgeResponse
      setKnowledgeItems(data.items)
      setTotal(data.total)
    } catch (err) {
      setError(err instanceof Error ? err.message : 'An error occurred')
    } finally {
      setIsLoading(false)
    }
  }
}
```

Gambar 4.31 Mengambil *KnowledgeSourceList*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.3.3.7 Integrasi API dan Utilitas

Implementasi integrasi API pada sistem *chatbot frontend* menggunakan pendekatan modular dan *scalable* dengan memisahkan *concerns* antara utilitas *HTTP client*, penanganan otentikasi, dan *endpoint* API spesifik.

- 1) Utilitas API (*src/lib/utills.ts* dan *src/lib/chat.ts*): File *src/lib/utills.ts* berisi fungsi-fungsi utilitas fundamental seperti *API_BASE_URL* yang mendefinisikan alamat dasar *backend* API, fungsi *handleApiResponse* untuk menangani respons HTTP secara konsisten (termasuk penanganan kesalahan), dan *getAuthHeader* untuk menyisipkan token JWT ke dalam *header* permintaan secara otomatis. Sementara itu, *src/lib/chat.ts* berisi fungsi-fungsi spesifik untuk berinteraksi dengan *endpoint* API terkait *chat*, seperti *sendMessage*, *getConversations*, *getConversation*, dan *deleteConversation*. Definisi tipe data seperti *Message*, *Source*, dan *Conversation* juga terdapat dalam *file* ini untuk memastikan konsistensi data antara *frontend* dan *backend*.
- 2) Manajemen Konfigurasi dan Penanganan Respons Terpusat: Pengelolaan konfigurasi API dilakukan melalui variabel lingkungan (*process.env.NEXT_PUBLIC_API_URL*) untuk kemudahan konfigurasi antar lingkungan pengembangan dan produksi. Fungsi *handleApiResponse* memastikan bahwa semua respons dari *backend* diperiksa statusnya, dan data JSON diekstrak atau kesalahan dilempar dengan pesan yang relevan, menyederhanakan logika penanganan respons di setiap panggilan API.

Secara keseluruhan, arsitektur *component-based* dengan pemisahan *concerns* yang jelas memungkinkan pengembangan *frontend* yang modular dan mudah dipelihara. Integrasi dengan *backend* API yang *seamless* melalui abstraksi *HTTP client* memberikan keandalan dalam komunikasi data. Optimalisasi performa dan fitur aksesibilitas, sebagian difasilitasi oleh Next.js dan Radix UI, memastikan bahwa aplikasi dapat diakses oleh berbagai jenis pengguna dengan perangkat yang beragam.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.4 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa seluruh komponen dalam sistem *chatbot* informasi skripsi bekerja sesuai dengan kebutuhan fungsional dan non-fungsional yang telah ditetapkan. Tujuan utama dari pengujian ini adalah untuk memverifikasi keandalan, kinerja, dan kegunaan sistem secara keseluruhan sebelum digunakan oleh pengguna akhir.

4.4.1 Deskripsi Pengujian

Pengujian yang digunakan dalam pengembangan sistem ini adalah *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE), *Retrieval-Augmented Generation Assessment* (RAGAs), *Black Box Testing*, *User Acceptance Test* (UAT), *System Usability Scale* (SUS), dan *Net Promoter Score* (NPS).

4.4.1.1 ROUGE

ROUGE adalah serangkaian metrik yang digunakan untuk mengevaluasi ringkasan otomatis dan terjemahan mesin dengan membandingkan teks yang dihasilkan oleh model (*candidate*) dengan satu atau lebih referensi teks sebagai *ground truth* dimana ada 12 data yang digunakan. Metrik ini bekerja dengan menghitung tumpang tindih unit linguistik seperti n-gram, urutan kata, dan pasangan kata terpanjang. Skor ROUGE juga berada dalam rentang [0, 1].

- ROUGE-N: Mengukur tumpang tindih *overlap* n-gram (urutan n kata) antara teks yang dihasilkan dan teks referensi. ROUGE-1 mengukur tumpang tindih kata tunggal (unigram), yang menunjukkan kesamaan konten. ROUGE-2 mengukur tumpang tindih pasangan kata (bigram), yang dapat menunjukkan tingkat kefasihan yang serupa.
- ROUGE-L: Mengukur *Longest Common Subsequence* (LCS) atau urutan kata umum terpanjang antara teks yang dihasilkan dan referensi. Metrik ini berguna untuk menilai kesamaan struktural dan koherensi kalimat secara keseluruhan.

Data referensi yang digunakan pada evaluasi hasil sistem *chatbot* ini dapat dilihat pada Lampiran 3 Data Referensi Evaluasi RAG dan *Response Chatbot*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.4.1.2 RAGAS

RAGAS adalah kerangka kerja yang dirancang khusus untuk mengevaluasi sistem RAG. Metrik ini menilai kinerja sistem dari tiga aspek utama: kesetiaan faktual (*faithfulness*), relevansi jawaban terhadap pertanyaan (*answer relevancy*), dan kualitas konteks yang diambil (*context relevancy & context recall*). Skor untuk setiap metrik berada pada rentang $[0, 1]$, di mana skor yang lebih tinggi menunjukkan kinerja yang lebih baik.

Faithfulness digunakan untuk mengukur sejauh mana jawaban yang dihasilkan oleh model didukung oleh informasi dari konteks yang diberikan. Skor *faithfulness* yang tinggi (mendekati 1) menandakan bahwa model tidak menghasilkan informasi yang tidak faktual. Sedangkan *answer relevancy* menilai apakah jawaban yang dihasilkan relevan dengan pertanyaan yang diajukan.

Context Precision & Context Recall: Context Precision mengukur rasio terhadap kerandoman dalam konteks yang diambil, memastikan bahwa informasi yang relevan mendominasi. *Context Recall* mengukur apakah semua informasi yang relevan dari ground truth dimana ada 12 data yang digunakan berhasil diambil oleh *retriever*. Data referensi yang digunakan pada evaluasi hasil sistem *chatbot* ini dapat dilihat pada Lampiran 3 Data Referensi Evaluasi RAG dan *Response Chatbot*.

4.4.1.3 Black Box Testing

Pengujian *Black Box* dilakukan untuk memastikan bahwa fitur-fitur utama dari sistem telah berjalan sesuai dengan kebutuhan fungsional. Pengujian disini dilakukan secara mandiri terlebih dahulu dan dielaborasi ketika UAT dengan Admin. Tabel berikut menunjukkan beberapa skenario pengujian:

Tabel 4.6 Pengujian Blackbox Testing

Kode	Fitur yang Diuji	Skenario Uji	Hasil yang Diharapkan
BT-1	<i>Chatbot</i> menjawab pertanyaan skripsi	Pengguna mengirim pertanyaan melalui antarmuka web	<i>Chatbot</i> merespons dengan jawaban kontekstual



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BT-2	Login	User Mencoba Login dengan kredensial yang valid	Sistem menampilkan halaman <i>chat</i>
BT-3	Pembaruan informasi oleh admin	Admin mengunggah dokumen skripsi baru	Sistem memproses dokumen dan update <i>knowledge</i>
BT-4	Melihat atau membuka document yang dilampirkan	Membuka file dari <i>knowledge resource</i>	Menampilkan file
BT-5	Manajemen log pengguna	Riwayat percakapan disimpan dan dapat diakses oleh pengguna	Log tersedia di <i>frontend</i> berdasarkan ID
BT-6	Delete Log Percakapan	Menghapus salah satu Log percakapan	Log percakapan yang dihapus menghilang

4.4.1.4 Pengujian *User Acceptance Test*

Pengujian *User Acceptance Test* (UAT) dilakukan untuk mengevaluasi sistem dari perspektif pengguna akhir, yaitu mahasiswa dan admin. Pengujian ini bertujuan untuk memastikan bahwa sistem *chatbot* informasi skripsi yang dikembangkan dapat digunakan secara nyata dan memenuhi ekspektasi pengguna dari segi fungsionalitas, kemudahan penggunaan, tampilan, serta keakuratan jawaban *chatbot*. Tabel 4.7 menentukan kategori dari hasil UAT dengan mahasiswa yang menggunakan pengukuran skala likert.

Tabel 4.7 Kriteria UAT

% Jumlah Skor	Kriteria
20% - 36%	Tidak Baik
37% - 52%	Kurang Baik
53% - 68%	Cukup
69% - 84%	Baik
85% - 100 %	Sangat Baik

Pengujian UAT dilakukan terhadap beberapa fitur utama sistem, antara lain *login*, interaksi *chatbot*, pengelolaan dokumen oleh admin, dan akses berdasarkan peran pengguna. Setiap penguji diberikan skenario uji dan diminta mengisi formulir evaluasi serta memberikan umpan balik. Pengujian UAT dilakukan terhadap 10



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

mahasiswa dan 1 admin, menggunakan kuesioner Likert 1–5 (mahasiswa) dengan 1 menandakan “Sangat Tidak Setuju”, 2 menandakan “Tidak Setuju”, 3 menandakan “Netral”, 4 menandakan “Setuju” dan 5 “Sangat Setuju” dan pengamatan *Sesuai/Gagal* (admin).

Tabel 4.8 Pertanyaan UAT User Mahasiswa Sistem *Chatbot*

Kode	Pertanyaan	Format Jawaban
UAT-1	Fitur registrasi dan <i>login</i> berjalan dengan baik	Skala Likert 1 – 5 1 = Sangat Tidak Setuju 2 = Tidak Setuju 3 = Netral 4 = Setuju 5 = Sangat Setuju
UAT-2	Tampilan web untuk registrasi dan <i>login</i> mudah dipahami	
UAT-3	Jawaban yang diberikan <i>chatbot</i> sesuai dengan kebutuhan saya	
UAT-4	Sumber Informasi dapat diakses dengan baik	
UAT-5	Alur keseluruhan fitur mudah dipahami	

Tabel 4.9 Pengujian Fungsional Dengan Admin TI TIK

Kode	Fitur yang Diuji	Langkah Uji	Hasil yang Diharapkan
UAT-A-1	Login Admin	Admin memasukkan kredensial valid	Sistem menampilkan dashboard admin
UAT-A-2	Unggah File Panduan Skripsi	Admin mengunggah file PDF	File tersimpan dan <i>chatbot</i> dapat mengakses isinya
UAT-A-3	Unggah File Panduan Skripsi	Admin mengunggah file IMAGE	File tersimpan dan <i>chatbot</i> dapat mengakses isinya



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

UAT-A-4	Unggah File Panduan Skripsi	Admin mengunggah file AUDIO	File tersimpan dan <i>chatbot</i> dapat mengakses isinya
UAT-A-5	Bertanya ke <i>Chatbot</i>	Mahasiswa mengajukan pertanyaan terkait skripsi	<i>Chatbot</i> memberikan jawaban akurat dan kontekstual
UAT-A-6	Peran Berdasarkan Role	Mahasiswa tidak bisa mengakses halaman admin	Sistem menolak akses tidak sah
UAT-A-7	Logout	Klik tombol logout di dashboard	Pengguna diarahkan ke halaman <i>login</i> kembali

4.4.1.5 System Usability Scale (SUS)

Pengujian SUS ditujukan untuk mengetahui tingkat kegunaan dari system ini. Semua pertanyaan dari SUS ini memiliki format jawaban yang sama yaitu menggunakan skala likert 1 sampai 5, Dengan 1 menandakan “Sangat Tidak Setuju”, 2 menandakan “Tidak Setuju”, 3 menandakan “Netral”, 4 menandakan “Setuju” dan 5 “Sangat Setuju”.

Tabel 4.10 Pertanyaan System Usability Scale

Kode	Pertanyaan
SUS-1	Saya berpikir akan menggunakan web ini lagi
SUS-2	Saya merasa web ini rumit untuk digunakan
SUS-3	Saya merasa web ini mudah untuk digunakan
SUS-4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan web ini
SUS-5	Saya merasa fitur-fitur web ini berjalan dengan semestinya
SUS-6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi) pada web ini
SUS-7	Saya merasa orang lain akan memahami cara menggunakan web ini dengan cepat

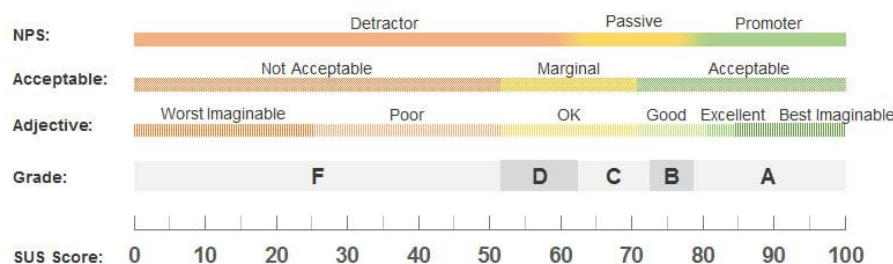


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

SUS-8	Saya merasa tidak ada hambatan dalam menggunakan web ini
SUS-9	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan web ini
SUS-10	Saya merasa web ini membingungkan

Dengan Gambar 4.31 menunjukkan bagaimana menginterpretasikan skor yang didapat (Sauro, 2018).



Gambar 4.31 Interpretasi Skor SUS

4.4.1.6 Net Promoter Score (NPS)

NPS bertujuan untuk mengetahui seberapa besar pengguna akhir merekomendasikan sistem ke orang lain. Dengan Pertanyaan pengujiannya yang dapat dilihat pada Tabel 4.11

Tabel 4.11 Pertanyaan Net Promoter Score

Kode	Pertanyaan	Skala Pengukuran
NPS-1	Saya akan merekomendasikan aplikasi ini kepada teman saya	Skala Likert 1 – 10 1 = Sangat Tidak Merekomendasikan 10 = Sangat Merekomendasikan

4.4.2 Hasil Pengujian

4.4.2.1 RAGAS

Hasil evaluasi menggunakan kerangka kerja RAGAS menunjukkan kinerja model yang kuat dalam aspek kesetiaan faktual dan relevansi, baik pada konteks maupun jawaban. Tabel 4.12 menyajikan rekapitulasi skor rata-rata untuk setiap metrik RAGAS.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Tabel 4.12 Hasil Skor RAGAS

Metrik	Skor Rata-Rata
Faithfulness	0.865
Answer Relevancy	0.802
Context Precision	0.797
Context Recall	0.806
Average RAGAS Score	0.8175

Skor Faithfulness sebesar 0.865 menunjukkan bahwa jawaban yang dihasilkan oleh model secara konsisten didukung oleh dokumen konteks yang diambil. Skor Answer Relevancy sebesar 0.802 mengindikasikan bahwa jawaban yang diberikan sangat relevan dengan pertanyaan pengguna.

Secara khusus, komponen retrieval menunjukkan kinerja yang baik dengan skor Context Recall sebesar 0.797, yang berarti sistem berhasil mengambil semua informasi yang relevan dari sumber data untuk setiap pertanyaan. Skor Context Precision sebesar 0.806 juga cukup tinggi, menandakan bahwa konteks yang diambil tidak hanya lengkap tetapi juga padat informasi dan kecil dari data yang tidak relevan. Secara keseluruhan, hasil RAGAS mengonfirmasi bahwa arsitektur RAG yang diimplementasikan efektif dalam mengambil konteks yang tepat dan menghasilkan jawaban yang andal dan relevan.

4.4.2.2 ROUGE

Evaluasi menggunakan ROUGE memberikan gambaran tentang kemiripan dalam struktural antara jawaban yang dihasilkan model dengan jawaban referensi. Tabel 4.20 menyajikan skor rata-rata untuk metrik ROUGE-1, ROUGE-2, dan ROUGE-L.

Tabel 4.13 Hasil Skor ROUGE

Metrik	Skor Rata-Rata
ROUGE-1	0.85



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

ROUGE-2	0.75
ROUGE-L	0.8
Rata Rata	0.8

Skor ROUGE-1 sebesar 0.85 menunjukkan adanya tumpang tindih kata kunci yang cukup baik antara jawaban model dan referensi, mengindikasikan bahwa konten utama berhasil disampaikan. Namun, skor ROUGE-2 yang lebih rendah senilai 0.75 menyiratkan bahwa pilihan frasa dan struktur kalimat model bervariasi dari jawaban referensi. Hal ini wajar terjadi pada model generatif yang memiliki kemampuan parafrase.

Skor ROUGE-L 0.8 yang berada di antara ROUGE-1 dan ROUGE-2 menunjukkan bahwa meskipun struktur kalimat tidak identik, alur logis dan informasi penting dalam jawaban model tetap sejalan dengan jawaban referensi. Hasil ROUGE secara kolektif menunjukkan bahwa model mampu menghasilkan jawaban yang akurat secara semantik, meskipun dengan variasi linguistik yang signifikan dibandingkan dengan teks referensi.

4.4.2.3 Black Box Testing

Hasil uji black box ditunjukkan pada Tabel 4.14, semua skenario telah berhasil dijalankan. Untuk *screenshot* hasil pengujian tampilan *user interface* nya dapat dilihat pada Lampiran 1. Hasil Blackbox Testing.

Tabel 4.14 Hasil Pengujian Black Box

Kode	Hasil yang Diharapkan	Pengamatan	Hasil
BT-1	<i>Chatbot</i> merespons dengan jawaban kontekstual	Sistem memberikan response dengan Bahasa alami dan menjawab sesuai konteks pertanyaan	Sesuai



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BT-2	Sistem menampilkan halaman <i>chat</i>	Sistem menampilkan halaman <i>chat</i>	Sesuai
BT-3	Sistem memproses dokumen dan update <i>knowledge</i>	Sistem mengupdate <i>knowledge</i> dengan tambahan data dokumen baru	Sesuai
BT-4	Menampilkan file	Sistem menampilkan preview dari file	Sesuai
BT-5	Log tersedia di <i>frontend</i> berdasarkan ID	Sistem memberikan record log Riwayat percakapan	Sesuai
BT-6	Log percakapan yang dihapus menghilang	Sistem menghapus/hilangkan Log percakapan	Sesuai

Berdasarkan hasil uji yang ditunjukkan pada Tabel 4.14, seluruh skenario yang berjumlah 6 menghasilkan Kesimpulan “Sesuai”. Sehingga untuk menghitung nilai efektifitas atau persentase kelayakan sistem (Karisma, Wahanani, & Nurlaili, 2021)

$$\left(\frac{\Sigma \text{Kesimpulan Sukses}}{\Sigma \text{Sampel data}} \right) \times 100$$

Gambar 4.32 Rumus Perhitungan Pengujian BlackBox

Maka berdasarkan data Tabel 4.14 didapat hasil yaitu:

$$\frac{6}{6} \times 100\% = 100\%$$

Kesimpulan dari pengujian black box testing dari hasil 100% adalah “Sangat Efektif” berdasarkan standar ukuran efektifitas sesuai acuan Litbang Depdagri (Putri, Ilhamsyah, & Mutiah, 2021).

4.4.2.4 User Acceptance Test

Hasil *User Acceptance Test* untuk mahasiswa dapat dilihat pada Tabel 4.15 dengan jumlah responden R=24, jumlah pertanyaan Q=5, skor maksimum per pertanyaan M=5. Sehingga skor ideal adalah R x Q x M = 600.

Tabel 4.15 Hasil User Acceptance Mahasiswa

Kode Responden	Q1	Q2	Q3	Q4	Q5
----------------	----	----	----	----	----



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

RUAT-M-1	5	4	4	5	4
RUAT-M-2	5	5	5	5	5
RUAT-M-3	5	5	5	5	5
RUAT-M-4	5	5	4	4	5
RUAT-M-5	5	5	5	5	4
RUAT-M-6	5	5	5	5	5
RUAT-M-7	5	5	5	5	5
RUAT-M-8	5	5	4	5	4
RUAT-M-9	5	5	3	3	4
RUAT-M-10	5	4	5	5	5
RUAT-M-11	5	5	4	4	5
RUAT-M-12	5	5	5	5	5
RUAT-M-13	4	4	4	4	4
RUAT-M-14	5	5	5	5	5
RUAT-M-15	5	5	5	5	5
RUAT-M-16	5	5	4	5	5
RUAT-M-17	5	5	5	4	5
RUAT-M-18	5	5	4	4	5
RUAT-M-19	4	3	4	3	4
RUAT-M-20	5	5	4	4	4
RUAT-M-21	5	5	5	5	5
RUAT-M-22	5	5	2	4	4
RUAT-M-23	4	4	5	4	4
RUAT-M-24	5	5	5	5	5
Jumlah	117	114	106	108	111

Rumus pengukuran skor dari data UAT yang didapat pada Tabel 4.15 adalah:

$$\% \text{ Skor Aktual} = \frac{\text{Skor Aktual}}{\text{Skor Ideal}} \times 100\%$$

Sehingga persentase skor yang didapat dari data pada Tabel 4.12 yaitu:

$$\% \text{ Skor Aktual} = \frac{556}{600} \times 100\%$$



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Persentase skor yang didapat dari hasil UAT dengan mahasiswa adalah 92,6% , sehingga kesimpulan bahwa UAT ini masuk dalam kriteria “Sangat Baik” berdasarkan Tabel 4.15 (Wulandari, Nofiyani, & Hasugian, 2023).

Tabel 4.16 Hasil User Acceptance Admin

Kode	Hasil yang Diharapkan	Pengamatan	Hasil
UAT-A-1	Sistem menampilkan dashboard admin	Sistem menampilkan dashboard admin	Sesuai
UAT-A-2	File tersimpan dan <i>chatbot</i> dapat mengakses isinya	File tersimpan dan <i>chatbot</i> dapat mengakses isinya	Sesuai
UAT-A-3	File tersimpan dan <i>chatbot</i> dapat mengakses isinya	File tersimpan dan <i>chatbot</i> dapat mengakses isinya	Sesuai
UAT-A-4	File tersimpan dan <i>chatbot</i> dapat mengakses isinya	File tersimpan dan <i>chatbot</i> dapat mengakses isinya	Sesuai
UAT-A-5	<i>Chatbot</i> memberikan jawaban akurat dan kontekstual	<i>Chatbot</i> memberikan jawaban akurat dan kontekstual	Sesuai
UAT-A-6	Sistem menolak akses tidak sah	Sistem menolak akses tidak sah	Sesuai



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

UAT-A-7	Logout	Pengguna diarahkan ke halaman <i>login</i> kembali	Sesuai
---------	--------	--	--------

Berdasarkan hasil uji yang ditunjukkan pada Tabel 4.16, seluruh skenario yang berjumlah 7 menghasilkan Kesimpulan “Sesuai”. Sehingga untuk menghitung nilai efektifitas atau persentase kelayakan sistem juga menggunakan rumus yang ada pada Gambar 4.31. Maka berdasarkan data pada Tabel 4.16 ini menghasilkan *persentase* skor yaitu :

$$\frac{7}{7} \times 100\% = 100\%$$

Kesimpulan dari pengujian UAT dengan admin dengan skor 100% adalah “Sangat Efektif” berdasarkan standar ukuran efektifitas sesuai acuan Litbang Depdagri (Putri, Ilhamsyah, & Mutiah, 2021).

4.4.2.5 System Usability Scale

Hasil SUS dari kuesioner uji coba dapat dilihat pada Tabel 4.17 dan Tabel 4.18.

Tabel 4.17 Hasil SUS

Kode	Responden	Q1	Q2	Q3	Q4	Q5
SUS-1	RSUS-1	4	2	4	2	5
SUS-2	RSUS-2	5	1	5	1	5
SUS-3	RSUS-3	4	2	4	2	4
SUS-4	RSUS-4	5	2	4	1	4
SUS-5	RSUS-5	4	1	5	1	4
SUS-6	RSUS-6	5	2	5	1	5
SUS-7	RSUS-7	5	1	5	1	5
SUS-8	RSUS-8	4	2	5	2	4
SUS-9	RSUS-9	3	4	3	4	3
SUS-10	RSUS-10	4	2	4	2	4
SUS-11	RSUS-11	5	1	5	1	5



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

SUS-12	RSUS-12	3	2	4	1	3
SUS-13	RSUS-13	4	2	4	1	4
SUS-14	RSUS-14	4	1	5	1	5

Tabel 4.18 Hasil SUS 2

Kode	Responden	Q6	Q7	Q8	Q9	Q10
SUS-1	RSUS-1	2	4	4	2	2
SUS-2	RSUS-2	1	4	5	1	1
SUS-3	RSUS-3	2	4	4	3	2
SUS-4	RSUS-4	1	5	5	4	2
SUS-5	RSUS-5	2	4	4	3	2
SUS-6	RSUS-6	1	5	5	2	2
SUS-7	RSUS-7	5	5	5	2	1
SUS-8	RSUS-8	2	4	5	2	2
SUS-9	RSUS-9	4	3	4	3	4
SUS-10	RSUS-10	3	3	4	2	2
SUS-11	RSUS-11	1	5	1	5	1
SUS-12	RSUS-12	4	4	4	1	1
SUS-13	RSUS-13	2	4	4	5	2
SUS-14	RSUS-14	1	4	5	1	1

Nilai SUS dapat dihitung dengan ketentuan skor pertanyaan berorientasi positif (untuk pertanyaan 1, 3, 5, 7, dan 8) dan kurang satu dari jawaban responden. Jawaban akan berkisar dari nol hingga empat. Kemudian skor pertanyaan berorientasi negatif (untuk pertanyaan 2, 4, 6, 9, dan 10) dan kurang jawaban responden dari lima. Setelah itu skor SUS didapat dari hasil penjumlahan skor setiap pertanyaan yang kemudian dikali 2,5. Hasil perhitungan SUS dirangkum pada Tabel 4.19

Tabel 4.19 Hasil Perhitungan SUS

Responden	Total Skor	Skor SUS (*2.5)
-----------	------------	-----------------



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

RSUS-1	31	77.5
RSUS-2	39	97.5
RSUS-3	29	72.5
RSUS-4	33	82.5
RSUS-5	32	80
RSUS-6	37	92.5
RSUS-7	35	87.5
RSUS-8	32	80
RSUS-9	17	42.5
RSUS-10	28	70
RSUS-11	32	80
RSUS-12	29	72.5
RSUS-13	28	70
RSUS-14	38	95
Total	792	1100
Rata-Rata Skor SUS	56,57	78,57

Karena rata-rata yang didapat untuk nilai SUS adalah 78,57 maka dapat dikatakan bahwa tingkat kegunaan (*usability*) sistem ini berada pada *grade* B atau kategori “Good”.

4.4.2.6 Net Promoter Score

Hasil NPS yang diperoleh dari kuesioner uji coba dapat dilihat pada Tabel 4.20 yang menampilkan jumlah responden berdasarkan tiap kategori.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Tabel 4.20 Hasil Perhitungan NPS

Skala	1	2	3	4	5	6	7	8	9	10
Jumlah jawaban							4	7	6	7
Kategori	Detractor						Passive		Promoter	
Persentase	0%						45,83%		54,16%	
NPS	54,16%									

Berdasarkan perhitungan nilai NPS yang tertera pada Tabel menunjukkan skor yang didapat dengan mengurangi persentase Promoter dengan Detractor sehingga skor nya adalah 54,16% atau termasuk kedalam kategori “*excellent*” (Carpenter, 2025). Hal ini menunjukkan bahwa mayoritas responden merasa puas dan bersedia menjadi promotor dari sistem ini.

**POLITEKNIK
NEGERI
JAKARTA**

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil implementasi dan pengujian sistem *chatbot* berbasis *Large Language Model* (LLM) sebagai sarana informasi skripsi yang telah diuraikan pada bab-bab sebelumnya, dapat ditarik kesimpulan bahwa sistem *chatbot* yang dirancang telah berhasil memenuhi tujuan yang telah diidentifikasi pada penelitian ini. Hal ini mencakup kemampuan sistem untuk memberikan respons otomatis terhadap pertanyaan terkait skripsi.

Sistem ini juga melakukan serangkaian pengujian, berdasarkan hasil pengujian *Black Box Testing* memperoleh skor *black box testing* dengan nilai 100% yang masuk kedalam kategori “Sangat Efektif”. Pengujian *User Acceptance Testing* (UAT) juga menunjukkan skor 92,6 % untuk UAT mahasiswa sehingga masuk dalam kategori “Sangat Baik”. UAT untuk admin menunjukkan skor 100% yang masuk dalam kategori “Sangat Baik”. Kemudian pengujian *System Usability Scale* rata-rata yang diperoleh untuk nilai SUS adalah 78,67 maka dapat dikatakan bahwa tingkat kegunaan (*usability*) sistem ini berada pada kategori “Good” berdasarkan standar interpretasi SUS. Kemudian *Net Promoter Score* memperoleh skor 54,16% termasuk kedalam kategori “*excellent*”. Lebih lanjut pada hasil evaluasi menggunakan RAGAS untuk RAG dan ROUGE yang mendapatkan skor rata-rata pada RAGAS 0.8175 dan rata-rata skor ROUGE 0.8 yang dari kedua skor disimpulkan bahwa alur RAG dan model berfungsi dengan sangat baik.

5.2 Saran

Meskipun sistem *chatbot* yang dikembangkan telah berhasil memenuhi tujuan penelitian, terdapat beberapa aspek yang dapat dikembangkan lebih lanjut untuk meningkatkan fungsionalitas, kinerja, dan pengalaman pengguna. Berikut adalah beberapa saran untuk pengembangan di masa mendatang:

1. Peningkatan Kemampuan Pemahaman Bahasa Alami Lanjutan: Meskipun telah menggunakan model LLM yang canggih seperti Llama-3.3-70b-



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

versatile ada ruang untuk peningkatan dalam menangani variasi pertanyaan yang lebih kompleks, bahasa informal yang sangat spesifik, atau istilah-istilah teknis kampus yang mungkin belum tercakup sepenuhnya

2. Optimalisasi Kinerja Unggah dan Pemrosesan Dokumen: Pengembangan pemecahan dokumen (*chunking*) serta proses *embedding*.

Dengan implementasi saran-saran di atas, diharapkan sistem *chatbot* ini dapat terus berkembang menjadi alat bantu informasi akademik yang semakin handal, cerdas, dan bermanfaat bagi seluruh sivitas akademika Politeknik Negeri Jakarta.





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR PUSTAKA

- Adobe. (n.d.). *Apa arti dari PDF?* From Adobe:
https://www.adobe.com/id_id/acrobat/about-adobe-pdf.html
- Alexander Tobias, N. Y. (2024, October). An LLM-Driven Chatbot in Higher Education for Databases and Information Systems. *IEEE Explorer*.
 doi:10.1109/TE.2024.3467912
- Assayed, S. K., Alkhatib, M., & Shaalan, K. (2024, April). A Systematic Review of Conversational AI Chatbots in Academic Advising. From
https://www.researchgate.net/publication/379449258_A_Systematic_Review_of_Conversational_AI_Chatbots_in_Academic_Advising
- Auffarth, B. (2024). Generative AI with LangChain. Build large language model (LLM) apps with python, ChatGPT, and other's LLM. In B. Auffarth, N. Katare, & K. Sharma (Eds.), *Generative AI with LangChain. Build large language model (LLM) apps with python, ChatGPT, and other's LLM* (p. 335). Birmingham, UK: Packt Publishing Ltd. Retrieved December, 2024 from
https://books.google.co.id/books?hl=en&lr=&id=cOrpEAAAQBAJ&oi=fnd&pg=PP1&dq=langchain&ots=VXsCx3oto&sig=VvFQHWrHUq1XaeW6pm86vdFicZk&redir_esc=y#v=onepage&q=langchain&f=false
- binus. (2024). *Metode Agile dalam Pengembangan suatu Software*. From binus.ac.id: <https://binus.ac.id/bekasi/2024/11/metode-agile-dalam-pengembangan-suatu-software/>
- Bunker, A. (2021). *What is NPS? The ultimate guide to boosting your Net Promoter Score*. From Qualtrics: <https://www.qualtrics.com/experience-management/customer/net-promoter-score/>
- Carpenter, A. (2025). *What is a good Net Promoter Score?* From Qualtrics: <https://www.qualtrics.com/experience-management/customer/good-net-promoter-score/>
- Content Square. (2024, October). *What is Net Promoter Score (NPS)? How to Calculate & Interpret*. From Content Square :
<https://contentsquare.com/guides/net-promoter-score/>



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- Digital Ocean. (2025). *Documentation of Digital Ocean*. From DigitalOcean:
<https://docs.digitalocean.com/products/>
- Dybedokken, O., & Nilsen, H. (2024). Agent-based chatbot in corporate environment. *University of Stavanger*. doi:3150399
- FastAPI. (2024). *Documentation FastAPI*. Retrieved December, 2024 from
<https://fastapi.tiangolo.com/>
- Google. (2024). *Machine Learning*. From Developers Google:
<https://developers.google.com/machine-learning/resources/intro-llms>
- GroqCloud. (2023). *Groq API Reference*. Retrieved December, 2024 from
<https://console.groq.com/docs/api-reference#chat-create>
- Holdsworth, J. (August 2024). *Natural Language Processing*. (C. Stryker, Editor)
 Retrieved Desember, 2024 from
<https://www.ibm.com/think/topics/natural-language-processing>.
- IBM. (2023). *UML Models and Diagrams*. From IBM:
<https://www.ibm.com/docs/en/rational-soft-arch/9.7.0?topic=diagrams-uml-models>
- Karisma, E. D., Wahanani, H. E., & Nurlaili, A. L. (2021). PENGUJIAN MENGGUNAKAN BLACK BOX BERBASIS. *Jurnal Informatika dan Sistem Informasi (JIFoSI)*, 277.
- Kurniawan, E., Nofriadi, & Nata, A. (2022). PENERAPAN SYSTEM USABILITY SCALE (SUS) DALAM. *Journal of Science and Social Research*, 4.
- LangChain. (2023). *LangChain Python API Reference*. Retrieved December, 2024 from https://python.langchain.com/api_reference/
- LucidChart. (2021). *UML Activity Diagram Tutorial*. From LucidChart:
<https://www.lucidchart.com/pages/uml-activity-diagram>
- Mistral. (2024). *Bienvenue to Mistral AI Documentation*. From docs.mistral.ai:
<https://docs.mistral.ai/>
- ms-johnalex, k. m. (2025). *Key concepts and considerations for building generative AI solutions*. (Microsoft) From Learn.Microsoft:
<https://learn.microsoft.com/en-us/azure/developer/ai/gen-ai-concepts-considerations-developers>



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- OpenAI. (2024). *OpenAI*. From OpenAI: <https://openai.com/>
- Oracle. (October 7, 2020). *What Is Chatbot?* Retrieved December, 2024 from <https://www.oracle.com/chatbots/what-is-a-chatbot/>
- Pinecone. (2024). *A Developer's Guide to Approximate Nearest Neighbor (ANN) Algorithms*. From pinecone: <https://www.pinecone.io/learn/a-developers-guide-to-ann-algorithms/>
- Pinecone. (2024). *Pinecone Docs*. From pinecone.io: <https://docs.pinecone.io/guides/get-started/overview>
- Putri, M. W., Ilhamsyah, & Mutiah, N. (2021). PENGUKURAN EFEKTIVITAS DAN EFISIENSI PENERAPAN OPEN DATA . *Jurnal Komputer dan Aplikasi* .
- Ragas. (2025, February). *Overview of Metrics*. From docs.ragas.com: <https://docs.ragas.io/en/latest/concepts/metrics/overview/>
- React. (2023). *The library for web and native user interfaces*. From React: <https://react.dev/>
- Sauro, J. (2018). *Interpret a SUS Score*. From MeasuringU: <https://measuringu.com/interpret-sus-score/>
- Solih, I. I. (2024). Pengembangan Chatbot layanan desa menggunakan algoritma Bidirectional Encoder Representations From Transformers pada Platform Whatsapp. *digilib.uinsgd.ac.id*.
- Supabase. (2024). *Documentation Supabase*. From Supabase: <https://supabase.com/>
- Topsakal, O., & Akinci, T. (2023, July). Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast. *ResearchGate*. doi:10.59287/icaens.1127
- Tripathi, R., & pinecone. (2023). *What is Similarity Search?* From pinecone: <https://www.pinecone.io/learn/what-is-similarity-search/>
- Twilio. (2024). *Twilio Docs*. From Twilio: <https://www.twilio.com/docs/whatsapp>
- Visual Studio Code. (2022). *Visual Studio Code Documentation*. From Visual Studio Code: <https://code.visualstudio.com/docs>
- Wulandari, Nofiyani, & Hasugian, H. (2023). USER ACCEPTANCE TESTING (UAT) PADA ELECTRONIC DATA. *JURNAL MAHASISWA ILMU KOMPUTER*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR RIWAYAT HIDUP PENULIS



Vidi Septri Argalus Mp

Lahir di Kutacane, 15 September 2003, anak ketiga dari tiga bersaudara. Lulus dari SDS St.Yoseph Lawe Desky tahun 2015, SMPS Panti Harapan Lawe Desky tahun 2018, SMAS Panti Harapan Lawe Desky pada tahun 2021. Saat ini sedang menempuh Pendidikan Sarjana Terapan di Politeknik Negeri Jakarta pada Jurusan Teknik Informatika dan Komputer di Program Studi Teknik Informatika. Penulis tertarik pada bidang *Web Development* dan *Artificial Intelligence*.

**POLITEKNIK
NEGERI
JAKARTA**

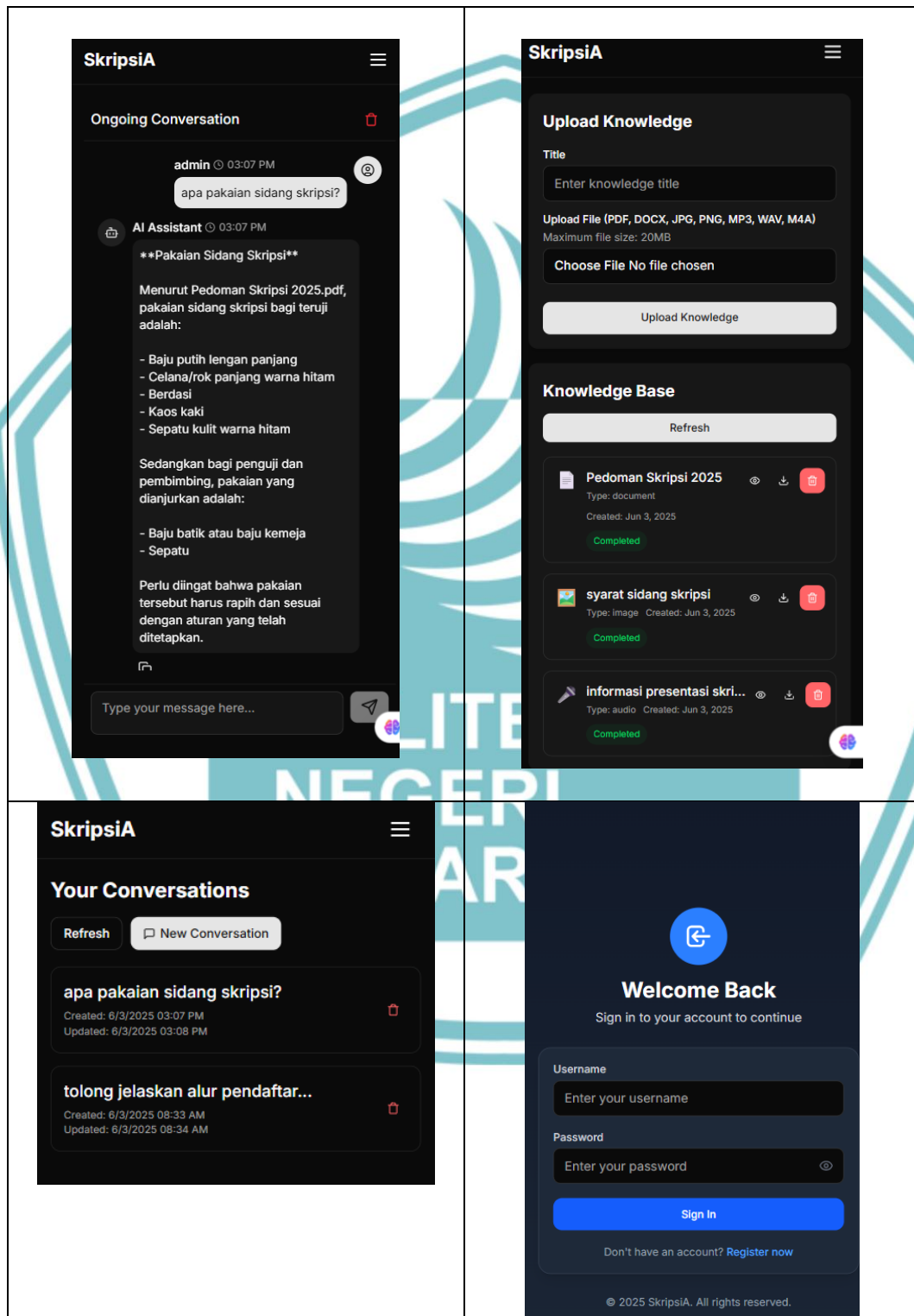


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LAMPIRAN

Lampiran 1 Hasil Blackbox Testing



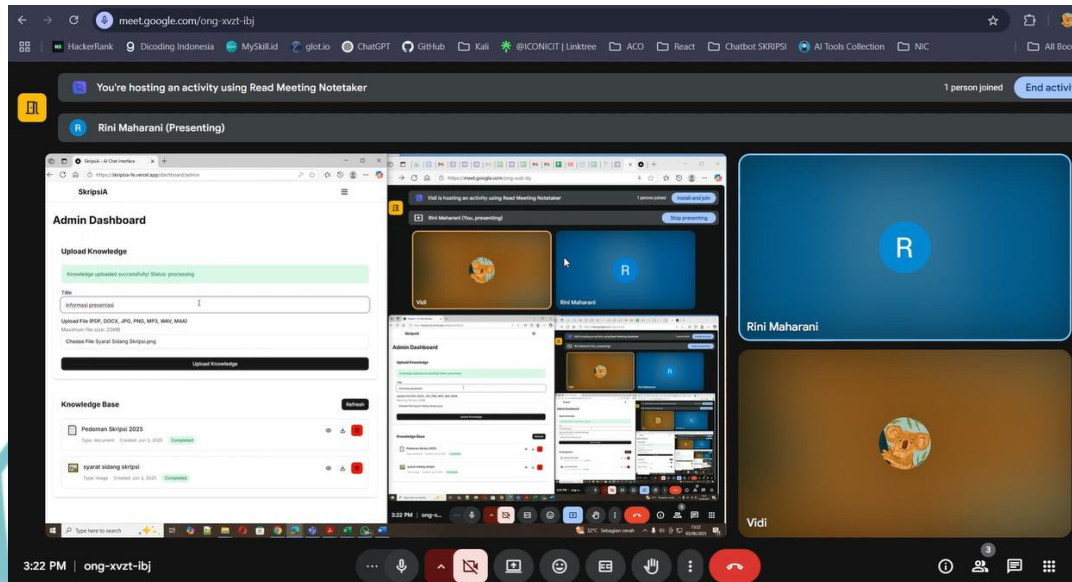


© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Lampiran 2 UAT dengan Admin Program Studi TI TIK



POLITEKNIK
NEGERI
JAKARTA



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Lampiran 3 Data Referensi Evaluasi RAG dan *Response Chatbot*

No	Pertanyaan	Jawaban Referensi	Konteks Pendukung
1	Apa yang dimaksud dengan skripsi menurut pedoman ini?	Skripsi adalah karya tulis ilmiah wajib dengan bobot 6 SKS yang disusun mahasiswa pada akhir masa studinya untuk memenuhi syarat kelulusan. Tujuannya adalah untuk melatih kemandirian, tanggung jawab ilmiah, dan kemampuan menerapkan metode yang tepat untuk memecahkan masalah.	"Skripsi adalah mata kuliah wajib dengan bobot 6 SKS seperti yang diatur dalam kurikulum yang berlaku, berupa karya tulis ilmiah yang disusun/ditulis mahasiswa pada akhir masa studinya untuk memenuhi persyaratan menyelesaikan pendidikan di Politeknik Negeri Jakarta.", "Skripsi merupakan suatu rangkaian kegiatan akademik yang bertujuan untuk menampilkan kompetensi yang dimiliki mahasiswa dalam bentuk penelitian tugas serta untuk melatih kemandirian dan tanggung jawab ilmiah mahasiswa mulai dari penyusunan rencana, pelaksanaan, evaluasi, hingga penulisan Laporan Skripsi.", "Dengan membuat Skripsi diharapkan mahasiswa memiliki cara berpikir khas untuk melihat, menggali, dan mengupas suatu masalah tertentu, terutama masalah yang terjadi di industri dan lingkungannya, serta menerapkan suatu metode yang tepat dalam rangka pencapaian hasil yang lebih optimal,



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

			baik dari segi teknis, ekonomis, maupun lingkungan."
2	Apa saja syarat untuk mengajukan proposal skripsi?	Syarat pengajuan proposal skripsi adalah memiliki IPK minimal 2.00 sampai dengan semester VII, berstatus sebagai mahasiswa aktif, sudah lulus semester 7, dan telah melakukan bimbingan minimal 3 kali yang tercatat di buku bimbingan.	"IPK sampai dengan semester VII ≥ 2.00 ", "Mahasiswa aktif semester berjalan ", "Mahasiswa menyertakan buku bimbingan yang sudah terisi hasil bimbingan (min 3x) dengan pembimbing dan disetujui oleh dosen pembimbing ", "Sudah lulus semester 7"
3	Berapa lama durasi sidang proposal skripsi?	Sidang proposal skripsi dilaksanakan dengan durasi 30 menit untuk setiap judul atau subjudul.	"Sidang Proposal Skripsi dilaksanakan dengan durasi 30 menit per Judul/Sub judul. ", "Sidang proposal akan terbagi menjadi 3 sesi. Sesi 1 (5-10 menit) adalah presentasi proposal oleh mahasiswa, sesi 2 (10-15 menit) adalah Tanya jawab bersama penguji dan sesi 3 adalah (5 menit) keputusan kelayakan proposal skripsi."
4	Bagaimana format penulisan umum untuk	Laporan skripsi diketik di atas kertas HVS A4 80 gram,	"Jenis : HVS ", "Berat : 80 gram ", "Ukuran: A4 (21,5 cm x 29,7 cm) ",



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

	laporan skripsi?	menggunakan huruf Times New Roman 12 poin dengan spasi 1,5. Margin kiri adalah 4 cm, sedangkan margin atas, bawah, dan kanan adalah 3 cm.	"Batas kiri : 4 cm dari tepi kertas", "Batas kanan : 3 cm dari tepi kertas", "Batas atas : 3 cm dari tepi kertas", "Batas bawah: 3 cm dari tepi kertas ", "Huruf menggunakan jenis huruf Times New Roman 12 poin (ukuran sebenarnya) dan diketik rapi (rata kiri kanan - justify).", "Pengetikan dilakukan dengan spasi 1,5 (Line spacing = 1.5 lines)."
5	Apa saja syarat untuk mendaftar sidang akhir skripsi?	Terdaftar sebagai mahasiswa aktif, IPK di semester VIII (tanpa nilai skripsi) minimal 2,00 tanpa nilai E, bimbingan minimal 10 kali, persetujuan dosen pembimbing, empat eksemplar laporan akhir, dan minimal 1 sertifikat lomba serta 4 sertifikat kegiatan ilmiah.	"Masih terdaftar sebagai mahasiswa Politeknik Negeri Jakarta aktif.", "Nilai IPK di semester VIII tidak termasuk nilai skripsi adalah $\geq 2,00$ tanpa nilai E.", "Menyerahkan bukti konsultasi bimbingan sekurang-kurangnya sepuluh kali bimbingan.", "Menyerahkan bukti lembar persetujuan pembimbing mengikuti sidang skripsi.", "Menyerahkan berkas laporan akhir skripsi sebanyak empat eksemplar (tidak perlu di jilid).", "Menyerahkan minimal 1 sertifikat lomba dan/atau sertifikat keikutsertaan kegiatan ilmiah(misal:



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

			kuliah umum, seminar, dll) minimal 4x."
6	Berapa jumlah penguji dalam sidang akhir skripsi?	Setiap sidang akhir skripsi diuji oleh tiga orang penguji, terdiri dari satu ketua sidang merangkap anggota dan dua orang anggota.	"Setiap sidang akhir skripsi diuji oleh tiga orang penguji, terdiri dari satu orang ketua sidang merangkap anggota, dua orang anggota dan dapat didampingi oleh pembimbing.", "Penguji sidang skripsi adalah penguji yang menguji proposal skripsi", "Dewan penguji tugas akhir terdiri atas 3 (tiga) orang penguji yang telah ditentukan oleh Kepala Program Studi."
7	Bagaimana ketentuan penulisan abstrak?	Abstrak ditulis maksimal 200 kata dalam satu paragraf, menggunakan huruf Times New Roman 11 poin, dicetak miring, dengan spasi tunggal. Disusun dalam bahasa Indonesia, dengan kata kunci di bagian bawah yang diurutkan alfabet.	"Abstrak merupakan ikhtisar suatu Skripsi yang memuat permasalahan, tujuan, metode penelitian, hasil, dan kesimpulan.", "Maksimum 200 kata dalam satu paragraf, diketik dengan tipe Times New Roman, 11 pt, dicetak miring (italic), spasi tunggal.", "Abstrak disusun dalam bahasa Indonesia.", "Di bagian bawah abstrak dituliskan kata kunci. Untuk Abstrak dalam bahasa Indonesia, kata kunci diberikan dalam bahasa Indonesia.", "Kata kunci diurut berdasarkan alphabet."



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

8	Apa saja isi dari bagian akhir proposal skripsi?	Daftar pustaka dan lampiran. Daftar pustaka harus mengacu pada Harvard Style, dan lampiran dapat berisi dokumen penunjang seperti data sheet atau gambar.	"Bagian akhir proposal terdiri dari daftar pustaka dan lampiran.", "Daftar pustaka berisi informasi tentang sumber pustaka yang telah dirujuk dalam tubuh tulisan.", "Daftar pustaka ditulis mengacu kepada Harvard Style.", "Lampiran dapat berupa dokumen penunjang topik Skripsi yakni data sheet, gambar, pekerjaan, dan lain-lain. "
9	Apa saja yang harus diserahkan setelah dinyatakan lulus sidang akhir?	Mahasiswa harus menyerahkan laporan skripsi yang sudah dijilid, soft file laporan untuk diunggah, artikel ilmiah format IEEE, video demo produk berdurasi 5-10 menit, dan alat atau program aplikasi yang menjadi milik PNJ. Penyerahan dilakukan paling lambat 10 hari kerja setelah	"Hasil rekayasa Alat pendukung /Program Aplikasi menjadi milik Politeknik Negeri Jakarta dan diserahkan kepada Jurusan.", "Laporan skripsi yang telah dijilid dan disahkan ketua jurusan serta alat pendukung/program diserahkan pada pihak-pihak terkait selambat-lambatnya sepuluh (10) hari kerja setelah syarat-syarat kelulusan terpenuhi. ", "Laporan Skripsi dalam bentuk soft file wajib diunggah dalam situs web institusi.", "Mengumpulkan artikel dari laporan skripsi sesuai dengan format artikel IEEE (https://s.pnj.ac.id/ArtikelIEEE).",



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

		syarat kelulusan terpenuhi.	"Mengumpulkan video demo produk/ hasil skripsi dengan durasi 5-10 menit dengan bentuk MP4 dalam bentuk link google drive"
10	Apakah skripsi boleh dikerjakan secara berkelompok?	Ya, boleh individu atau kelompok (minimal 2 orang), bahkan lintas prodi. Penentuan anggota berdasarkan tingkat kesulitan, waktu, biaya, dan ruang lingkup.	"Jumlah mahasiswa peserta Skripsi minimal satu orang, atau boleh berupa kelompok 2 orang atau lebih, dan dapat dilakukan dengan lintas program studi.", "Penentuan jumlah anggota kelompok disesuaikan dengan:", "a. Tingkat kesulitan", "b. Lama waktu pengerjaan Skripsi", "c. Biaya", "d. Ruang lingkup"
11	Bagaimana aturan berpakaian saat sidang skripsi?	Mahasiswa wajib memakai kemeja putih lengan panjang, celana/rok panjang warna hitam, berdasi, memakai kaos kaki, dan sepatu kulit hitam.	"Berpakaian rapih baju putih lengan panjang, celana/rok panjang warna hitam, berdasi dan memakai kaos kaki serta sepatu kulit warna hitam."
12	Apa saja ketentuan mengenai referensi untuk skripsi?	Judul skripsi harus didukung minimal 5 referensi ilmiah (3 jurnal nasional, 2	"Pemilihan judul Skripsi harus memiliki latar belakang yang kuat dengan menyertakan minimal 5 referensi ilmiah terkait dengan judul Skripsi yang dipilih (minimal 3 artikel



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

	publikasi internasional terindeks Scopus), maksimal terbit 5 tahun terakhir, dan wajib menyitasi publikasi dosen JTIK.	dari jurnal nasional dan 2 artikel dari publikasi internasional terindeks scopus, contoh: IEEEExplore, Elsevier).", "Referensi tambahan dapat diambil dari buku teks, white paper, majalah ilmiah dan wajib dicantumkan pada daftar pustaka.", "Seluruh referensi baik referensi utama maupun tambahan diambil maksimal 5 tahun terakhir dan diwajibkan mensitasi dari publikasi dosen JTIK."
--	--	---

**POLITEKNIK
NEGERI
JAKARTA**