



© 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



**IMPLEMENTASI SISTEM WAWANCARA KERJA
BERBASIS *RETRIEVAL-AUGMENTED GENERATION*
UNTUK OTOMATISASI PERTANYAAN DAN
JAWABAN**

SKRIPSI

**POLITEKNIK
NEGERI
JAKARTA**

AHMAD ZAINUL MUFID

2207412045

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

POLITEKNIK NEGERI JAKARTA

2026



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



**IMPLEMENTASI SISTEM WAWANCARA KERJA
BERBASIS *RETRIEVAL-AUGMENTED GENERATION*
UNTUK OTOMATISASI PERTANYAAN DAN
JAWABAN**

SKRIPSI

**Dibuat untuk melengkapi Syarat – Syarat yang Diperlukan
untuk Memperoleh Diploma Empat Politeknik**

AHMAD ZAINUL MUFID

2207412045

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2026



SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini:

Nama : Ahmad Zainul Mufid
NIM : 2207412045
Jurusan/Program Studi : T. Informatika dan Komputer/Teknik Informatika
Judul Skripsi : Implementasi Sistem Wawancara Kerja Berbasis Retrieval-Augmented Generation untuk Otomatisasi Pertanyaan

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.

Depok, 15 Juli 2026

Yang membuat pernyataan

Ahmad Zainul Mufid

NIM.2207412045

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

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 : Ahmad Zainul Mufid

NIM : 2207412045

Program Studi : Teknik Informatika

Judul Skripsi : Implementasi Sistem Wawancara Kerja Berbasis Retrieval-Augmented Generation untuk Otomatisasi Pertanyaan dan Jawaban

Telah diuji oleh tim penguji dalam Sidang Skripsi pada hari Selasa, Tanggal 07 Bulan Juli Tahun 2026 dan dinyatakan **LULUS**.

Disahkan oleh

Pembimbing I : Euis Oktavianti, S.Si., M.T.I.

Penguji I : Dr. Dewi Yanti Liliana, S.Kom., M.Kom.

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

Penguji III : Chairul Fikri, M.Kom.

Mengetahui:

Jurusan Teknik Informatika dan Komputer

Ketua



Dr. Anita Hidayati, S.Kom., M.Kom.

NIP. 19790832003122003



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

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas segala rahmat, karunia, dan berkah-Nya sehingga penulis dapat menyelesaikan skripsi ini dengan baik sebagai salah satu syarat untuk memperoleh gelar Sarjana Terapan di Politeknik Negeri Jakarta.

Dalam proses penyusunan skripsi ini, penulis memperoleh banyak dukungan, bantuan, dan bimbingan dari berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

- a. Ibu Anita Hidayati, S.Kom., M.Kom., selaku Ketua Jurusan Teknik Informatika dan Komputer, atas dukungan dan kemudahan administrasi selama proses penyusunan skripsi.
- b. Ibu Euis Oktavianti, S.Si., M.T.I., selaku dosen pembimbing, atas arahan, bimbingan, serta masukan yang diberikan selama penyusunan skripsi.
- c. Kedua orang tua tercinta, Ayah dan Mamah, atas doa, kasih sayang, dukungan moral, serta bantuan yang senantiasa diberikan.
- d. Teman-teman dan seluruh pihak yang telah memberikan dukungan serta membantu dalam penyelesaian skripsi ini.

Sebagai penutup, penulis menyadari bahwa skripsi ini masih memiliki keterbatasan dan belum sepenuhnya sempurna. Namun demikian, penulis berharap penelitian ini dapat memberikan manfaat, menambah wawasan, serta menjadi kontribusi bagi pengembangan ilmu pengetahuan dan referensi bagi penelitian selanjutnya.

Depok, 15 Juli 2026

Penulis

Ahmad Zainul Mufid



© 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 bertanda tangan di bawah ini:

Nama : Ahmad Zainul Mufid

NIM : 2207412045

Jurusan/Program Studi : T. 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:

Implementasi Sistem Wawancara Kerja Berbasis *Retrieval-Augmented Generation* Untuk Otomatisasi Pertanyaan

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 sebenarnya.

Depok, 15 Juli 2026

Yang membuat pernyataan

Ahmad Zainul Mufid

NIM.2207412045



IMPLEMENTASI SISTEM WAWANCARA KERJA BERBASIS RETRIEVAL-AUGMENTED GENERATION UNTUK OTOMATISASI PERTANYAAN DAN JAWABAN

ABSTRAK

Wawancara kerja merupakan tahapan krusial bagi pencari kerja, khususnya kalangan mahasiswa dan fresh graduate, yang membutuhkan persiapan matang serta latihan simulasi yang realistis. Karena simulasi konvensional sering terbatas pada pertanyaan statis, penelitian ini bertujuan mengimplementasikan sistem simulasi wawancara otomatis berbasis suara memanfaatkan kecerdasan buatan. Sistem ini memadukan *Large Language Model* (LLM), metode *Retrieval-Augmented Generation* (RAG), dan basis data vektor ChromaDB. Interaksi suara didukung teknologi *Speech recognition* (Whisper API) untuk mentranskripsi jawaban dan *Text-to-Speech* (Edge TTS) untuk menghasilkan suara pewawancara AI. Penilaian jawaban dihitung secara matematis menggunakan TF-IDF dan *Cosine Similarity* untuk mengukur kemiripan jawaban kandidat dengan jawaban referensi, sementara LLM berperan sebagai reasoning engine penyusun narasi umpan balik dan laporan evaluasi. Performa sistem diuji melalui empat tahap: pengujian kecerdasan menggunakan RAGAs menunjukkan relevansi konteks yang tinggi, *human evaluation* oleh dua orang pakar menunjukkan 10 dari 15 sampel umpan balik dinilai valid, pengujian fungsionalitas *Black Box* membuktikan seluruh fitur berjalan dengan baik, serta evaluasi *System Usability Scale* (SUS) memperoleh skor rata-rata 91 (Grade A, *Excellent*) dan *Net Promoter Score* (NPS) mencapai skor 90.

Kata Kunci: *Large Language Model*, Otomatisasi Pertanyaan, *Retrieval-Augmented Generation*, *Speech Recognition*, *Cosine Similarity*.

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 ISI

SURAT PERNYATAAN BEBAS PLAGIARISME	i
LEMBAR PENGESAHAN	ii
KATA PENGANTAR	iii
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS	iv
ABSTRAK	v
DAFTAR ISI	vi
DAFTAR GAMBAR	viii
DAFTAR TABEL	ix
DAFTAR LAMPIRAN	x
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan dan Manfaat	3
1.4.1 Tujuan	3
1.4.2 Manfaat	3
1.5 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA	5
2.1 Wawancara Kerja	5
2.1.1 Generate Pertanyaan Otomatis	5
2.1.2 Aplikasi Wawancara Kerja	5
2.2 Software Development Life Cycle (SDLC)	6
2.2.1 Waterfall	6
2.3 Cross Industry Standard Process–Data Mining (CRISP-DM)	9
2.3.1 Business Understanding (Pemahaman Bisnis)	9
2.3.2 Data Understanding (Pemahaman Data)	10
2.3.3 Data Preparation (Persiapan Data)	10
2.3.4 Modelling	10
2.3.5 Evaluation (Evaluasi)	13
2.3.6 Deployment (Penerapan)	16
2.4 Penelitian Terdahulu	16

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



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

BAB III METODE PENELITIAN.....	16
3.1 Rancangan Penelitian.....	16
3.2 Tahapan Penelitian.....	17
3.3 Objek Penelitian	19
BAB IV HASIL DAN PEMBAHASAN	20
4.1 Analisis Kebutuhan.....	20
4.1.1 Kebutuhan Dataset	20
4.1.2 Kebutuhan Perangkat Keras.....	21
4.1.3 Kebutuhan Perangkat Lunak.....	21
4.1.4 Kebutuhan Fungsional	23
4.1.5 Kebutuhan Non-Fungsional.....	24
4.2 Perancangan Sistem	24
4.2.1 Use Case Diagram	25
4.2.2 Activity Diagram	26
4.2.3 Sequence Diagram.....	31
4.2.4 Class Diagram.....	39
4.2.5 Perancangan Antarmuka	40
4.3 Implementasi Sistem.....	43
4.3.1 Implementasi Pengumpulan Basis Pengetahuan	43
4.3.2 Implementasi Retrieval-Augmented Generation (RAG)	49
4.3.3 Implementasi Penilaian Jawaban Kandidat	56
4.3.4 Implementasi Speech-to-Text Menggunakan Whisper.....	72
4.3.5 Implementasi Antarmuka Pengguna	75
4.4 Pengujian	79
4.4.1 Deskripsi Pengujian.....	79
4.4.2 Prosedur Pengujian.....	81
4.4.3 Data Hasil Pengujian	85
4.4.4 Analisis Data Pengujian	109
BAB V PENUTUP	112
5.1 Kesimpulan.....	112
5.2 Saran	113
DAFTAR PUSTAKA	114
DAFTAR RIWAYAT HIDUP PENULIS	118
LAMPIRAN	119



DAFTAR GAMBAR

Gambar 2.1 Cara Kerja RAG	11
Gambar 3.1 Rancangan Penelitian	16
Gambar 4.1 Use Case Diagram	25
Gambar 4.2 Activity Diagram Proses Pendaftaran Akun	27
Gambar 4.3 Activity Diagram Proses Login Akun	28
Gambar 4.4 Activity Diagram Proses Memilih Pekerjaan	29
Gambar 4.5 Activity Diagram Proses Sesi Wawancara.....	30
Gambar 4.6 Activity Diagram Melihat Riwayat Wawancara.....	31
Gambar 4.7 Sequence Diagram Proses Pendaftaran Akun.....	32
Gambar 4.8 Sequence Diagram Proses Login Akun.....	33
Gambar 4.9 Sequence Diagram Proses Memilih Pekerjaan.....	34
Gambar 4.10 Sequence Diagram Proses Menampilkan Pertanyaan	35
Gambar 4.11 Sequence Diagram Proses Menjawab Pertanyaan.....	36
Gambar 4.12 Sequence Diagram Proses Melihat Hasil Evaluasi.....	37
Gambar 4.13 Sequence Diagram Melihat Riwayat Wawancara	38
Gambar 4.14 Class Diagram Sistem InterviewMate	39
Gambar 4.15 Rancangan Halaman Login.....	40
Gambar 4.16 Rancangan Halaman Register	41
Gambar 4.17 Rancangan Halaman Memilih Pekerjaan	41
Gambar 4.18 Rancangan Halaman Sesi Wawancara	42
Gambar 4.19 Rancangan Halaman Hasil Evaluasi.....	43
Gambar 4.20 Transformasi Data	44
Gambar 4.21 Penggabungan Dataset	45
Gambar 4.22 Penghapusan Data Duplikat	46
Gambar 4.23 Penambahan Metadata.....	47
Gambar 4.24 Pembentukan Teks Embedding	48
Gambar 4.25 Penyimpanan Data ke Vector Database	49
Gambar 4.26 Fungsi Adaptive Question.....	50
Gambar 4.27 Penggabungan Konteks.....	51
Gambar 4.28 Generate Pertanyaan.....	53
Gambar 4.29 Proses Perhitungan Kecocokan Jawaban	54
Gambar 4.30 Pengembalian Objek Hasil Evaluasi	55
Gambar 4.31 Generate Laporan Akhir.....	56
Gambar 4.32 Antarmuka Pengguna Halaman Login	76
Gambar 4.33 Antarmuka Pengguna Halaman Pendaftaran	76
Gambar 4.34 Antarmuka Pengguna Halaman Interview Room	77
Gambar 4.35 Antarmuka Pengguna Halaman Sesi Wawancara	78
Gambar 4.36 Antarmuka Pengguna Halaman Laporan	78
Gambar 4.37 Antarmuka Pengguna Halaman Laporan Riwayat Transkripsi	79
Gambar 4.38 Responden Status Pekerjaan	107

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



DAFTAR TABEL

Tabel 1 Penelitian Terdahulu.....	16
Tabel 2 Kebutuhan Dataset.....	20
Tabel 3 Kebutuhan Perangkat Keras Pengembangan Model	21
Tabel 4 Kebutuhan Perangkat Keras Pengembangan Web	21
Tabel 5 Kebutuhan Perangkat Lunak Pengembangan Model	22
Tabel 6 Kebutuhan Perangkat Lunak Pengembangan Web	23
Tabel 7 Kebutuhan Fungsional.....	23
Tabel 8 Kebutuhan Non-Fungsional	24
Tabel 9 Data Pertanyaan dan Jawaban.....	57
Tabel 10 Case Folding	58
Tabel 11 Tokenizing	59
Tabel 12 Filtering	59
Tabel 13 Stemming	60
Tabel 14 Daftar Term Hasil Preprocessing.....	60
Tabel 15 Hasil Perhitungan TF.....	62
Tabel 16 Hasil Perhitungan DF	64
Tabel 17 Hasil Perhitungan IDF	66
Tabel 18 Hasil Perhitungan TF-IDF	68
Tabel 19 Hasil Perhitungan Cosine Similarity	70
Tabel 20 Konfigurasi Parameter Akuisisi Audio	73
Tabel 21 Paramater Whisper	74
Tabel 22 Prosedur Pengujian Black Box Testing.....	82
Tabel 23 Persiapan Kuesioner SUS	83
Tabel 24 Pertanyaan NPS	85
Tabel 25 Hasil Pengujian Evaluasi RAGAs	85
Tabel 26 Hasil Pengujian Human Evaluation.....	86
Tabel 27 Hasil Pengujian Black Box Testing.....	101
Tabel 28 Hasil Pengujian SUS	108
Tabel 29 Hasil Perhitungan SUS	108
Tabel 30 Hasil Pengujian NPS	109

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



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

DAFTAR LAMPIRAN

Lampiran 1 Wawancara	119
Lampiran 2 Hasil Kuesioner SUS	121
Lampiran 3 Hasil Kuesioner NPS	125
Lampiran 4 Pengujian Aplikasi	125





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 I PENDAHULUAN

1.1 Latar Belakang

Tingginya persaingan dalam proses rekrutmen kerja berdampak langsung pada rendahnya peluang pelamar untuk lolos tahap wawancara. Berdasarkan laporan statistik rekrutmen *Apollo Technical* (2024) mengungkapkan fakta penting bahwa hanya sekitar 20% pelamar yang berhasil dipanggil wawancara, dan dari jumlah tersebut, hanya 17% yang akhirnya menerima tawaran kerja. Data ini menekankan bahwa wawancara merupakan fase seleksi paling ketat dalam rekrutmen. Banyak pelamar gugur di tahap ini bukan hanya karena kurangnya kualifikasi teknis, tetapi juga akibat ketidakmampuan mengomunikasikan kompetensi mereka dengan baik di bawah tekanan (Giffar et al., 2024).

Fenomena ini mengindikasikan bahwa kurangnya akses pelamar terhadap sarana simulasi wawancara yang menyebabkan rendahnya kesiapan mental dan materi. Akibatnya, pelamar seringkali kurang memiliki ekspektasi yang jelas tentang jenis pertanyaan yang akan mereka hadapi (Giffar et al., 2024). Permasalahan ini semakin krusial pada bidang pekerjaan yang menuntut spesialisasi tinggi seperti Teknologi Informasi, di mana proses wawancara umumnya melibatkan pertanyaan teknis yang mendalam dan spesifik (Brown et al., 2025). Tanpa platform latihan yang mampu mensimulasikan kondisi wawancara secara nyata, pelamar akan kesulitan mengevaluasi kualitas jawaban mereka sendiri secara objektif.

Berkaitan dengan permasalahan yang telah diuraikan, banyak penelitian telah membahas pemanfaatan teknologi kecerdasan buatan dalam otomatisasi proses wawancara melalui berbagai pendekatan dan metode. Penelitian oleh Pathak et al, (2025) menjelaskan sistem berbasis *multi-agent* yang memanfaatkan *Large Language Model (LLM)* seperti GPT-4 untuk menjalankan seleksi wawancara teknis secara otomatis dan melakukan evaluasi kandidat secara otomatis.

Meskipun penggunaan LLM memberikan efisiensi dan kemampuan pemahaman bahasa yang tinggi (Saha et al., 2024), tetapi memiliki kelemahan



© 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

seperti model diterapkan secara mandiri tanpa konteks eksternal. Model ini juga sering mengalami halusinasi, sehingga model ini memiliki kecenderungan menghasilkan informasi yang terdengar menakutkan namun tidak relevan dengan data yang ada (Lata, 2025).

Dalam mengurangi halusinasi, terdapat tiga pendekatan utama yang diterapkan, yaitu *Prompt Engineering*, *Fine Tuning*, dan *Retrieval-Augmented Generation (RAG)*. Teknik *Prompt Engineering* sering kali dibatasi oleh pada kapasitas memori, yang menyebabkannya proses analisis menjadi tidak konsisten pada dokumen resume yang panjang atau kompleks (Yang & Yang, 2023). Selanjutnya, Metode *Fine-Tuning* juga memiliki keterbatasan yang kurang efektif untuk kasus rekrutmen yang dinamis. Hal ini dikarenakan *Fine-Tuning* menuntut proses pelatihan ulang setiap kali terdapat data pelamar baru, sehingga membutuhkan biaya dan sumber daya komputasi yang sangat besar (Balaguer et al., 2023).

Oleh karena itu, metode *Retrieval-Augmented Generation (RAG)* diusulkan sebagai solusi efektif untuk mengurangi tingkat halusinasi AI. Metode RAG bekerja dengan cara menggabungkan kemampuan generatif model bahasa dan mekanisme pencarian data dari basis pengetahuan (*knowledge base*) eksternal (Hu et al., 2025). Melalui pendekatan ini, sistem tidak lagi bergantung pada dokumen statis, melainkan dapat mengekstrak topik keahlian kandidat secara langsung dari sesi awal pengenalan untuk dijadikan kueri pencarian kriteria pekerjaan. Dengan menjadikan data dari basis pengetahuan sebagai landasan fakta, pertanyaan wawancara yang dihasilkan menjadi sangat terpersonalisasi, akurat, dan minim halusinasi dibandingkan jika hanya menggunakan LLM standar (Tohir et al., 2024). Berdasarkan permasalahan dan peluang solusi tersebut, penelitian ini membutuhkan aplikasi simulasi wawancara kerja berbasis web yang menggunakan metode *Retrieval-Augmented Generation (RAG)* untuk otomatisasi personalisasi pertanyaan. Sistem ini dirancang menggunakan *Large Language Model (LLM)* *LLaMA-3.3-70B* yang dioptimalkan.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka perumusan masalah dalam penelitian ini adalah bagaimana mengimplementasikan metode *Retrieval-*



© 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

Augmented Generation dalam melakukan otomatisasi pertanyaan pada sistem wawancara kerja?

1.3 Batasan Masalah

Terdapat beberapa batasan masalah dalam penelitian ini yang perlu diperhatikan, sebagai berikut:

1. Sistem yang dikembangkan difokuskan sebagai media latihan wawancara kerja, bukan sebagai sistem seleksi resmi oleh perusahaan.
2. Sistem dikhususkan untuk simulasi wawancara kerja pada bidang teknologi informasi.
3. Sistem menerapkan metode RAG untuk mengambil konteks dari dokumen.
4. Interaksi suara menggunakan teknologi *whisper* untuk *Speech-to-Text* dan *edge text-to-speech*.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Tujuan dari penelitian ini adalah mengimplementasikan sistem wawancara kerja menggunakan metode *Retrieval-Augmented Generation* untuk otomatisasi proses pembuatan pertanyaan.

1.4.2 Manfaat

Berikut adalah beberapa manfaat yang diharapkan dari penelitian ini:

1. Memudahkan pelamar dalam melatih kesiapan wawancara kerja
2. Memberikan gambaran mengenai potensi pertanyaan
3. Memberikan kesempatan kepada wawancara kerja dalam membangun rasa percaya diri
4. Memberikan umpan balik bagi *interviewee*

1.5 Sistematika Penulisan

Dalam penulisan laporan ini penulis membuat suatu sistematika penulisan yang berdasarkan beberapa bab yang dimana masing-masing bab dapat diuraikan sebagai berikut

BAB I PENDAHULUAN

Berisi mengenai latar belakang, rumusan masalah, batasan, tujuan, dan manfaat, serta prosedur penulisan laporan.



© 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

BAB II TINJAUAN PUSTAKA

Berisi kumpulan diskusi tentang teori dasar dan komentar para ahli yang mendukung topik yang dibahas

BAB III METODE PENELITIAN

Berisi penjelasan mengenai metode yang akan digunakan, termasuk rancangan penelitian, tahapan penelitian, objek penelitian, sumber data, metode pengumpulan data, teknik analisis data, jadwal pelaksanaan, dan detail biaya.

BAB IV HASIL DAN PEMBAHASAN

Berisi penjelasan mengenai pemaparan dan analisis hasil implementasi RAG dan web simulasi wawancara kerja yang meliputi analisis kebutuhan, perancangan, implementasi, pengujian serta hasil analisis pengujian.

BAB V PENUTUP

Berisi penjelasan mengenai kesimpulan akhir dari penelitian dilengkapi dengan saran untuk penelitian selanjutnya.



**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

BAB II TINJAUAN PUSTAKA

2.1 Wawancara Kerja

Dalam proses rekrutmen saat ini, Wawancara kerja memegang peranan krusial sebagai penentu utama dalam menghubungkan antara pelamar dengan peluang karir mereka (Budiasningrum et al., 2021). Tahapan ini berfungsi sebagai sarana validasi pelamar untuk memastikan bahwa kandidat memiliki kompetensi yang sesuai dengan kebutuhan (Innayah et al., 2021). Evaluasi dalam proses ini sangatlah komprehensif, pelamar tidak hanya menjawab ketepatan jawaban yang diberikan, melainkan juga mengobservasi cara penyampaianya.

2.1.1 Generate Pertanyaan Otomatis

Pertanyaan merupakan sebuah ungkapan yang dirancang untuk menggali informasi spesifik pada lawan bicara. Dalam konteks wawancara kerja, pertanyaan tidak hanya berfungsi sebagai alat komunikasi verbal, melainkan bertindak sebagai instrumen utama untuk mencapai kompetensi yang dimiliki kandidat, mulai dari pengalaman teknis, pola perilaku, hingga kemampuan penyelesaian masalah (Wijoyo, 2022). Seiring berkembangnya teknologi khususnya di bidang *Natural Language Processing (NLP)*, proses pembuatan pertanyaan mulai beralih ke arah otomatisasi melalui metode *Automatic Question Generation (AQG)*.

AQG merupakan sebuah sistem komputasi yang mampu menganalisis input berupa teks pernyataan seperti deskripsi dalam resume dan mengubahnya menjadi kalimat tanya yang valid secara sintaksis dan *semantic*. Tujuan utama dari implementasi AQG dalam sistem wawancara adalah untuk mengurangi pewawancara dalam membuat pertanyaan secara manual dalam penyusunannya, sekaligus memastikan bahwa setiap pertanyaan yang dihasilkan benar-benar terpersonalisasi berdasarkan data unik yang dimiliki oleh setiap pelamar (Guo et al., 2021).

2.1.2 Aplikasi Wawancara Kerja

Saat ini, Aplikasi wawancara kerja telah merubah dari sekadar media komunikasi video menjadi sistem cerdas berbasis *Artificial Intelligence (AI)*.



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 memanfaatkan teknologi *Large Language Model (LLM)* dan *Retrieval-Augmented Generation (RAG)*, sistem dapat menjalankan simulasi wawancara teknis dan memberikan evaluasi otomatis secara mendalam. Implementasi nyata dari kemajuan ini terlihat pada kehadiran inovasi seperti Zara (Yazdani et al., 2025) yang dirancang untuk mengotomatisasi proses seleksi, serta pengembangan chatbot simulasi berbasis RAG (Anwar et al., 2025) yang menyediakan simulasi latihan interaktif.

2.2 Software Development Life Cycle (SDLC)

Kerangka kerja standar untuk rekayasa perangkat lunak yang dikenal sebagai *Software Development Life Cycle (SDLC)* digunakan secara sistematis untuk merancang, membangun, dan memelihara sistem informasi (Musthofa & Haryono, 2023). Pada penelitian ini, penulis menggunakan model *waterfall* karena model ini dapat digunakan untuk membuat sistem aplikasi dalam waktu yang relatif singkat. Kemudian, alur *waterfall* terstruktur untuk tahapan perencanaan, analisis, desain implementasi, pengujian, dan pemeliharaan. (Musthofa & Haryono, 2023).

2.2.1 Waterfall

Model *waterfall* merupakan sebuah metode pengembangan perangkat lunak yang menggunakan pendekatan sistematis dan sekuensial (Zen et al., 2022). Dalam penerapannya, setiap fase pengembangan harus diselesaikan secara tuntas dan tervalidasi terlebih dahulu sebelum proses dapat berlanjut ke tahap selanjutnya. Berikut ini tahapan metode *waterfall*.

1. Tahap Perencanaan

Pada tahap perencanaan, pengembang melakukan identifikasi kebutuhan dan batasan sistem. Proses ini melibatkan pengumpulan informasi strategis yang diperoleh melalui metode wawancara, survei, atau diskusi (Zen et al., 2022). Data yang terkumpul kemudian dianalisis untuk menetapkan ruang lingkup proyek, yang selanjutnya diklasifikasikan menjadi kebutuhan fungsional dan non-fungsional sebagai berikut.

a. Fungsional

Sebuah daftar fitur dan kemampuan teknis yang harus dimiliki oleh sistem disebut kebutuhan fungsional. Semua proses sistem, dari input data hingga pemrosesan 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

output yang dihasilkan, dijelaskan secara rinci dalam spesifikasi ini. (Wahyuni, 2023).

b. Non-Fungsional

Sebuah sistem yang tidak memenuhi persyaratan fungsional seperti kinerja, keamanan, skalabilitas, dan usabilitas disebut sebagai kebutuhan non-fungsional. (Wahyuni, 2023).

2. Tahap Desain

Pada tahap ini, pengembang membuat arsitektur sistem sesuai dengan permintaan pengguna (Binangkit et al., 2023). Penelitian ini akan menggunakan *Unified Modelling Language (UML)* untuk menerapkan desain sistem pengembangan. Beberapa contoh UML yang akan digunakan dalam penelitian ini adalah *Use Case Diagrams*, *Activity Diagrams*, *Class Diagrams*, dan *User Interface Design*.

a. Use Case Diagram

Use Case Diagram menunjukkan fitur apa saja yang akan digunakan oleh sistem. Diagram ini dapat membantu dalam menyusun persyaratan sistem dan merancang skenario pengujian untuk semua fitur yang ada. (Binangkit et al., 2023).

b. Activity Diagram

Activity Diagram menunjukkan aktivitas sistem yang ada dan menunjukkan urutan proses alur aktivitas dari awal sampai akhir. Diagram ini juga menunjukkan keputusan logika yang mungkin terjadi saat sistem berjalan. (Binangkit et al., 2023).

c. Sequence Diagram

Sequence Diagram menunjukkan aktivitas interaksi antar objek secara lebih rinci dalam suatu sistem. Diagram ini digunakan untuk membantu memahami kebutuhan sistem, mendokumentasikan proses, serta memvisualisasikan skenario ketika sistem dijalankan (Rohmanto & Setiawan, 2022).

d. Class Diagram

Class biasanya di indikasikan untuk menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan atribut pada sistem, bersamaan dengan layanan untuk memanipulasi fungsi tersebut. (Binangkit et al., 2023).



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

e. User Interface (UI)

Tujuan perancangan UI adalah untuk menciptakan antarmuka visual yang menghubungkan pengguna dengan sistem. Ini dimaksudkan untuk memungkinkan interaksi yang efektif dan memenuhi kebutuhan fungsional yang telah dirancang.

3. Tahap Implementasi

Setelah desain sistem selesai dirancang, tahap implementasi dimulai, di mana pengembang mulai mengubah desain ke dalam kode program. (Binangkit et al., 2023). Tujuannya untuk membangun sistem agar menjadi aplikasi berbasis website. Pengembangan antarmuka dibangun menggunakan React dan Typescript, sedangkan logika sisi server dan pemrosesan RAG diimplementasikan dengan bahasa program Python dengan basis data PostgreSQL. Selain itu, fitur interaksi suara diintegrasikan menggunakan model *Whisper* dan *Edge Text-to-Speech*.

4. Tahap Pengujian

Tahap berikutnya adalah pengujian. Di sini, setiap komponen sistem akan diuji secara menyeluruh untuk mengetahui seberapa baik kinerjanya. (Angelina et al., 2024). Pengujian tidak hanya berfokus pada fungsionalitas website, tetapi juga mencakup evaluasi akurasi model AI dalam Menyusun pertanyaan wawancara serta pengalaman pengguna saat mengoperasikan aplikasi.

a. Black Box Testing

Metode pengujian fungsional yang dilakukan tanpa melihat struktur kode program disebut *Black Box Testing*. Pengujian ini hanya menguji pada antarmuka pengguna, di mana penguji hanya memvalidasi apakah setiap fitur memberikan output yang sesuai dengan input yang dimasukkan (Praniffa et al., 2023).

b. System Usability Scale

System Usability Scale (SUS) merupakan evaluasi yang digunakan untuk mengukur seberapa kegunaan suatu aplikasi (Kholifah et al., 2023). Instrumen SUS menggunakan teknik kuesioner yang terdiri dari sepuluh pertanyaan. Responden memberikan tanggapan dengan mengisi kuesioner pada skala likert 1 sampai 5 dengan karakteristik unik berupa stuktur pertanyaan yang bersifat selang-seling, di mana pernyataan bernomor ganjil memuat kalimat positif yang mendukung aplikasi, sedangkan pernyataan bernomor genap memuat kalimat negatif. Hasil akhir dari perhitungan SUS berupa skor tunggal dengan rentang 0 hingga 100, 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 mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

memudahkan pengembang untuk menyimpulkan apakah sistem layak digunakan atau tidak (Kholifah et al., 2023).

c. Net Promoter Score (NPS)

Net Promoter Score (NPS) merupakan sebuah metode yang digunakan untuk mengukur sejauh mana tingkat loyalitas pengguna terhadap suatu produk atau layanan setelah mereka menggunakannya. Pengukuran menggunakan NPS dinilai lebih akurat dan kuat dibandingkan hanya sekedar mengukur tingkat retensi atau niat pembelian ulang (Tarigan et al., 2024). Hal ini dikarenakan retensi pengguna tidak selalu mencerminkan loyalitas sejati, pengguna bisa saja bertahan menggunakan suatu produk hanya karena faktor kebiasaan, kenyamanan sesaat, atau kurangnya alternatif produk lain di pasaran.

Penilaian metrik NPS dilakukan dengan mengajukan satu pertanyaan utama kepada pengguna mengenai seberapa besar kemungkinan mereka akan merekomendasikan produk tersebut kepada orang lain. Pertanyaan tersebut dijawab menggunakan skala angka 0 hingga 10.

Berdasarkan rentang nilai yang diberikan, responden diklasifikasikan ke dalam tiga kelompok. Kelompok pertama adalah *Detractor* (skor 0–6), yaitu pengguna yang merasa tidak puas dan tidak ingin merekomendasikan produk. Kelompok kedua adalah *Passive* (skor 7–8), di mana pengguna merasa cukup puas namun masih rentan berpindah jika ada alternatif produk yang lebih baik. Terakhir adalah kelompok *Promoter* (skor 9–10), yang menandakan bahwa pengguna merasa sangat puas dan memiliki keinginan yang sangat tinggi untuk merekomendasikan produk tersebut secara aktif.

2.3 Cross Industry Standard Process–Data Mining (CRISP-DM)

Metode CRISP-DM merupakan sebuah metodologi yang digunakan untuk menentukan sebuah standar dari proses data mining atau kaitannya. Proses ini terdiri dari enam tahap, pemahaman bisnis, pemahaman data, persiapan data, modeling, analisis, dan penerapan (Rianti et al., 2023)

2.3.1 Business Understanding (Pemahaman Bisnis)

Pemahaman Bisnis merupakan tahapan awal yang berfokus pada penggalian informasi mendalam mengenai kondisi dan tujuan bisnis untuk disusun menjadi rencana proyek (Rianti et al., 2023).



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

2.3.2 Data Understanding (Pemahaman Data)

Pemahaman Data merupakan tahapan yang mengumpulkan dataset mentah. Data tersebut kemudian dianalisis kualitas dan strukturnya untuk memastikan informasi didalamnya relevan sebelum diproses lebih lanjut (Rianti et al., 2023).

2.3.3 Data Preparation (Persiapan Data)

Persiapan Data merupakan tahapan lanjutan dalam mengolah data mentah menjadi dataset akhir yang siap digunakan. Proses ini mencakup pembersihan, transformasi, serta penyesuaian format data untuk memastikan kualitas input bagi sistem (Rianti et al., 2023).

2.3.4 Modelling

Modelling merupakan tahapan pemilihan teknis analisis dan diterapkan untuk membangun model yang dapat memecahkan masalah bisnis (Rianti et al., 2023). Dalam penelitian ini, tahap modelling bertujuan membangun sistem otomatisasi wawancara untuk menghasilkan pertanyaan menggunakan bahasa program python dan metode *Retrieval-Augmented Generation* (RAG).

a. Python

Python merupakan sebuah bahasa pemrograman yang sangat populer, karena banyak kegunaannya dalam berbagai jenis, seperti membangun logika sistem backend, pengembangan model data, algoritma machine learning, hingga penerapan data mining (Junaidi et al., 2023).

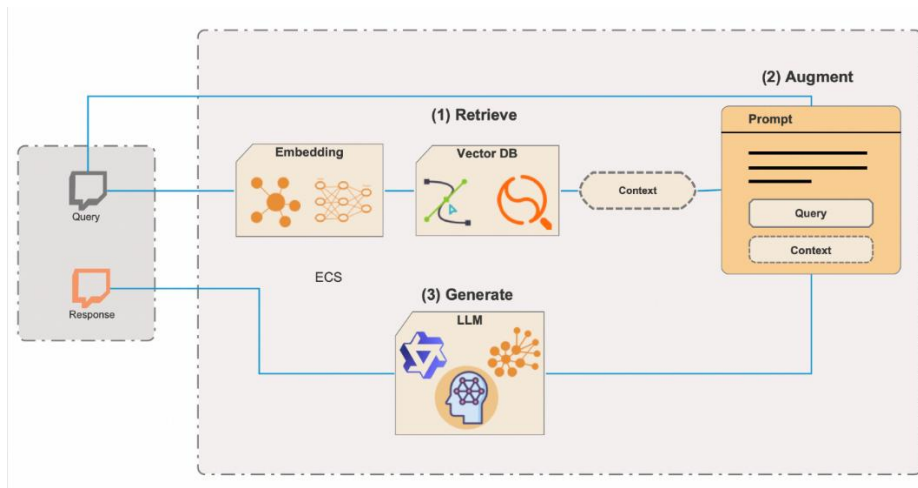
b. Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) merupakan metode yang dirancang untuk meningkatkan kualitas luaran model bahasa dengan cara menggabungkan kemampuan dalam mengambil informasi (*retrieval*) dan pembuatan teks (*generation*) (Albert & Voutama, 2025). Dalam penelitian ini, RAG bekerja dengan mengambil informasi referensi dari basis data *vektor* eksternal untuk dijadikan konteks sebelum respons dibuat. Sehingga RAG dapat memastikan bahwa pertanyaan wawancara yang dihasilkan sistem benar-benar fakta. Gambar 2.1 menunjukkan cara RAG bekerja.



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 2.1 Cara Kerja RAG
(Sumber: www.alibabacloud.com)

Gambar 2.1 mengilustrasikan tahapan kerja sistem RAG dalam merespons masukan pengguna yang terbagi menjadi tiga proses utama, yaitu *retrieve*, *augment*, dan *generate*. Proses pertama dimulai ketika terdapat masukan berupa *query* yang dalam sistem ini berasal dari teks hasil transkripsi jawaban lisan pelamar saat memberikan respons awal. *Query* tersebut kemudian diproses menggunakan *embedding* untuk diubah menjadi representasi *vektor* numerik, lalu sistem akan mencari tingkat kedekatan *vektornya* di dalam *vector database*. Hasil dari pencarian ini berupa *context* yang paling relevan dengan keahlian atau topik yang diucapkan pelamar, sehingga sistem dapat bekerja secara dinamis tanpa perlu bergantung pada dokumen fisik.

Setelah informasi atau konteks yang relevan berhasil didapatkan, sistem akan melanjutkan ke tahap berikutnya yaitu proses *augment* atau penggabungan. Pada tahap ini, informasi pendukung yang telah disaring tersebut disisipkan dan dipadukan bersama dengan *query* asli pelamar ke dalam sebuah templat instruksi. Proses penggabungan ini bertujuan untuk memperkaya insting dan pengetahuan model dengan data eksternal yang faktual sebelum memberikan respons akhir.

Pada tahap akhir atau *generate*, *Large Language Model* (LLM) bertugas untuk memproses prompt yang telah diperkaya oleh konteks tersebut secara cerdas. LLM kemudian mengesekusi instruksi untuk menghasilkan luaran akhir berupa *response*. Luaran ini dapat berupa pertanyaan wawancara lanjutan yang dinamis maupun



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

laporan evaluasi yang akurat, objektif, dan tetap berlandaskan pada data yang sesuai dengan performa serta kapasitas pelamar.

c. Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) merupakan sebuah metode statistik yang digunakan untuk mengevaluasi tingkat kepentingan suatu kata (*term*) di dalam sebuah dokumen yang tergabung dalam sebuah kumpulan dokumen (S et al., 2023). Metode ini berfokus pada dua elemen utama yaitu frekuensi kemunculan kata dalam satu dokumen tertentu (TF) dan seberapa umum atau langkanya kata tersebut di seluruh kumpulan dokumen (IDF).

Setelah nilai TF dan IDF untuk setiap term diperoleh, tahap selanjutnya adalah menghitung bobot akhir TF-IDF. Bobot ini merupakan hasil perkalian antara nilai TF (yang menunjukkan seberapa sering suatu term muncul pada sebuah dokumen) dengan nilai IDF (yang menunjukkan seberapa khas term tersebut di dalam korpus). Semakin tinggi nilai bobot TF-IDF suatu term pada sebuah dokumen, maka semakin penting dan semakin merepresentasikan term tersebut terhadap dokumen itu. Formulasi matematis untuk menghitung nilai bobot TF-IDF mengacu pada persamaan berikut:

$$TF - IDF(t, d) = TF(t, d) \times IDF(t)$$

Keterangan:

- $TF-IDF(t, d)$ = bobot term t pada dokumen d
- $TF(t, d)$ = nilai Term Frequency term t pada dokumen d
- $IDF(t)$ = nilai Inverse Document Frequency term t

d. Cosine similarity

Cosine similarity merupakan sebuah algoritma yang digunakan untuk menghitung kesamaan antara dua buah objek utama yang dinyatakan dalam dua buah vektor (Siswipraptini et al., 2023). Secara matematis, algoritma ini mengukur tingkat kesamaan dengan cara menghitung nilai kosinus dari sudut di antara kedua vektor tersebut pada ruang multidimensi. Dalam penelitian ini, *Cosine similarity* diimplementasikan pada dua proses utama, yaitu pada tahapan penarikan konteks dari basis data vektor (*vector database*) dan sebagai salah satu parameter hibrida (*hybrid*) dalam mengevaluasi tingkat akurasi jawaban pelamar terhadap kunci jawaban referensi. Rumus yang digunakan untuk *Cosine similarity* sebagai berikut:



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

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

Dimana:

A = vektor pertama yang digunakan dalam proses perbandingan kemiripan

B = vektor kedua yang digunakan dalam proses perbandingan kemiripan

A · B = hasil *dot product* (perkalian titik) antara vektor A dan vektor B

|A| = nilai panjang atau *magnitudo* dari vektor A

|B| = nilai panjang atau *magnitudo* dari vektor B

|A| × |B| = hasil perkalian *magnitudo* vektor A dan vektor B yang digunakan sebagai faktor normalisasi pada perhitungan *Cosine similarity*

2.3.5 Evaluation (Evaluasi)

Evaluasi merupakan tahapan pengujian yang esensial untuk mengukur kualitas dari model yang dibangun. Pada penelitian ini, evaluasi dilakukan melalui dua pendekatan utama, yaitu evaluasi otomatis menggunakan framework *Retrieval-Augmented Generation Assessment* (RAGAs) dan evaluasi kualitatif melalui human evaluation (evaluasi oleh manusia).

RAGAs merupakan sebuah kerangka kerja evaluasi standar yang dirancang secara khusus untuk mengukur kinerja sistem *Retrieval-Augmented Generation* (RAG) (Kharisma et al., 2025). Kerangka kerja ini memanfaatkan pendekatan *Large Language Model* (LLM) sebagai penilai untuk mengevaluasi dua komponen utama dalam arsitektur RAG, yaitu komponen konteks yang diambil (*retrieval*) dan komponen teks yang dihasilkan (*generation*). Adapun metrik RAGAs yang digunakan pada tahap ini meliputi *Faithfulness*, *Answer Relevance*, dan *Context Precision*.

Meskipun RAGAs mampu memberikan hasil pengukuran yang objektif, human evaluation digunakan untuk memvalidasi hasil penalaran (*reasoning*) yang dihasilkan oleh sistem. Evaluasi oleh manusia ini bertujuan untuk memastikan bahwa respons atau jawaban yang diberikan oleh model tidak hanya memenuhi standar metrik secara teknis, tetapi juga logis, natural, dan relevan ketika digunakan secara nyata oleh pengguna akhir. Kehadiran human evaluation menutupi keterbatasan penilaian otomatis dengan memberikan perspektif yang lebih peka terhadap nuansa bahasa dan konteks.



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

a. Faithfulness

Metrik *Faithfulness* digunakan untuk mengukur tingkat konsistensi *faktual* antara jawaban yang dihasilkan oleh sistem dengan konteks dokumen yang diambil (Pratama & Avianto, 2025). Evaluasi dilakukan untuk mencocokkan setiap klaim dalam jawaban terhadap konteks yang ada. Nilai akhir dihitung berupa rasio 0 hingga 1, di mana nilai yang mendekati angka 1 mengindikasikan jawaban akurat dan terhindari dari halusinasi (Pratama & Avianto, 2025). Untuk menghitung skor *Faithfulness* sebagai berikut:

$$Faithfulness = \frac{\text{Jumlah klaim yang terverifikasi}}{\text{Total jumlah klaim dalam jawaban}}$$

Keterangan:

1. Jumlah klaim yang terverifikasi adalah pernyataan dalam jawaban sistem yang dapat dibuktikan kebenarannya pada dokumen sumber.
2. Total jumlah klaim dalam jawaban adalah semua pernyataan yang berhasil diidentifikasi dalam jawaban yang dihasilkan oleh sistem.

b. Answer Relevance

Metrik *Answer Relevance* digunakan untuk mengukur seberapa tepat respon sistem dalam menjawab pertanyaan yang diajukan oleh pengguna berdasarkan kesamaan semantik (Pratama & Avianto, 2025). Evaluasi dilakukan dengan cara membuat pertanyaan hipotesis dari jawaban sistem, kemudian menghitung rata-rata kemiripan *vektor* antara pertanyaan hipotesis tersebut dengan pertanyaan asli. Nilai skor yang mendekati 1 mengindikasikan bahwa jawaban sistem sangat relevan dan secara langsung menjawab kebutuhan informasi pengguna. Untuk menghitung skor *Answer Relevance* sebagai berikut:

$$Answer\ Relevance = \frac{1}{N} \sum_{i=1}^n \cosine_{similarity}(E_q, E_{sqi})$$

Keterangan:

1. N adalah jumlah pertanyaan hipotesis yang dibuat
2. E_q adalah embedding dari pertanyaan asli pengguna
3. E_{sqi} adalah embedding dari pertanyaan hipotesis ke-i yang dihasilkan dari jawaban



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. Cosine Similarity adalah fungsi kesamaan kosinus antara dua vektor

c. Context Precision

Context Precision merupakan sebuah metrik evaluasi yang berfokus pada akurasi pemeringkatan dalam hasil pencarian. Metrik ini mengukur sejauh mana sistem mampu menempatkan potongan informasi di urutan teratas, bukan sekadar menemukannya. Nilai *context precision* yang tinggi menunjukkan bahwa sistem berhasil memprioritaskan dokumen relevan agar lebih mudah diproses oleh model. Perhitungan metrik ini menggunakan pendekatan *average precision* dengan rumus sebagai berikut

$$\text{Context Precision}@k = \frac{\sum_{k=1}^K (\text{Precision}@k \times u_k)}{\text{Total Relevant Items}}$$

Keterangan:

1. K adalah jumlah total potongan konteks yang diambil oleh sistem
2. $\text{Precision}@k$ adalah nilai presisi pada peringkat ke-k
3. u_k adalah nilai biner 1 atau 0, bernilai 1 jika item pada peringkat ke-k, dan 0 jika tidak
4. Total Relevant Items adalah jumlah total item relevan yang terdapat dalam kumpulan konteks yang diambil

d. Human Evaluation

Human Evaluation merupakan pendekatan penilaian yang melibatkan manusia sebagai evaluator untuk mengukur kualitas keluaran dari suatu sistem bahasa. Berbeda dengan evaluasi otomatis yang menggunakan metrik kuantitatif seperti BLEU atau ROUGE, metode tersebut sering kali memiliki korelasi yang lemah dengan penilaian manusia, khususnya pada tugas-tugas yang bersifat terbuka. Oleh karena itu, evaluasi manusia menjadi sangat penting untuk melengkapi kekurangan tersebut dengan menitik beratkan pada aspek-aspek kualitatif yang lebih kompleks dan bersifat subjektif, seperti relevansi, kejelasan, koherensi, kealamian, dan kepuasan pengguna (Babu & Krishnan, 2025).

Dalam konteks penelitian ini, human evaluation digunakan untuk memvalidasi hasil penalaran yang dihasilkan oleh sistem. Evaluasi oleh manusia ini bertujuan untuk memastikan bahwa respons atau jawaban yang diberikan oleh model tidak hanya memenuhi standar metrik secara teknis, tetapi juga logis, natural, dan relevan ketika



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

digunakan secara nyata oleh pengguna akhir. Kehadiran human evaluation secara efektif menutupi keterbatasan penilaian otomatis dengan memberikan perspektif yang lebih peka terhadap nuansa bahasa dan kesesuaian konteks percakapan.

2.3.6 Deployment (Penerapan)

Penerapan merupakan tahapan terakhir yang bertujuan untuk mengintegrasikan model ke dalam aplikasi website. Pada tahap ini, layanan backend dibangun menggunakan *Flask REST API* sebagai komunikasi dengan antarmuka website, yang kemudian di deploy ke server berbasis *Virtual Private Server* agar dapat diakses oleh pengguna.

2.4 Penelitian Terdahulu

Tabel 1 Penelitian Terdahulu

Judul	Masalah	Metode	Hasil
Retrieval-Augmented Generation (RAG) and Memory Systems for HR and Enterprise AI (Lata, 2025)	Kelemahan pada Large Language Model (LLM) standar yang sering menghasilkan halusinasi dan tidak memiliki akses aman terhadap data rahasia perusahaan	Menerapkan arsitektur RAG yang dikombinasikan dengan sistem memori dan basis data vektor untuk menghubungkan AI secara langsung dengan dokumen internal HRD	Penerapan RAG mampu mengurangi tingkat halusinasi serta menghasilkan jawaban yang akurat, transparan, dan sesuai dengan fakta
MockLLM: A Multi-Agent Behavior Collaboration Framework for Online Job Seeking and Recruiting (Sun & Lin, 2025)	Sistem rekrutmen daring saat ini masih kaku, karena hanya mencocokkan teks resume dengan deskripsi pekerjaan. Sehingga gagal menangkap kemampuan interaksi dan komunikasi kandidat	Metode MockLLM yang berperan sebagai agen ganda dimana satu agen AI berperan sebagai pewawancara dan agen lainnya berperan sebagai pelamar	MockLLM mampu menghasilkan dialog wawancara yang jauh realistis dan beragam, serta meningkatkan akurasi dalam memprediksi kecocokan kerja
EzInterview: To Improve Job Interview Performance with Mock Interview Generator (Li et al., 2025)	Latihan wawancara yang berkualitas sulit diakses dan mahal. Selain itu, sistem otomatis yang ada sering gagal menghasilkan pertanyaan yang benar-benar relevan dengan pengalaman spesifik pelamar	Mengembangkan EzInterview. Sistem ini tidak langsung membuat pertanyaan, melainkan mengekstrak kecocokan antara resume dan job description untuk menghasilkan pertanyaan	Sistem ini mampu menghasilkan pertanyaan wawancara yang jauh lebih bermakna dan akurat dibandingkan model bahasa biasa
Utilizing Retrieval-Augmented Generation in	LLM sering kesulitan dalam memberikan jawaban yang akurat	Menerapkan Teknik RAG untuk menghubungkan LLM	Penerapan RAG mampu meningkatkan kinerja pemrosesan bahasa alami, membuat jawaban AI



© 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

Large Language Models to Enhance Indonesian Language NLP (Tohir et al., 2024)	untuk pertanyaan spesifik, karena terbatasnya data pelatihan berkualitas tinggi dalam bahasa Indonesia	dengan sumber data eksternal	menjadi lebih relevan, akurat, dan sesuai konteks
ICCA-RAG: Intelligent Customs Clearance Assistant Using Retrieval-Augmented Generation (Hu et al., 2025)	Dokumen regulasi dan kepabeanan (Bea Cukai) sangat kompleks dan dinamis. Penggunaan LLM biasa sering menyebabkan halusinasi karena gagal memahami konteks aturan yang spesifik	Mengembangkan ICCA-RAG, menggunakan pendekatan pencarian hybrid dan mekanisme pemeringkatan ulang	Sistem ini mencapai tingkat akurasi yang sangat tinggi dalam menjawab pertanyaan regulasi yang rumit, dan mampu menghilangkan halusinasi pada domain

Berdasarkan penelitian terdahulu pada tabel 1, khususnya penelitian Sun & Lin, (2025) dan Li et al., (2025), terlihat adanya pengembangan pesat pada sistem wawancara otomatis. namun demikian, penelitian-penelitian tersebut masih bergantung pada model berbasis generatif murni yang sering kali kurang akurat secara faktual. Kekurangan ini dapat diatasi melalui penerapan metode *Retrieval-Augmented Generation (RAG)*, yang terbukti mampu memvalidasi kebenaran data pada domain sensitive seperti HRD (Lata, 2025). Hal ini diperkuat oleh penelitian (Tohir et al., 2024) yang menyatakan bahwa RAG efektif mengatasi keterbatasan LLM dalam memproses instruksi spesifik berbahasa Indonesia. Selain itu pada penelitian (Hu et al., 2025) dapat membuktikan bahwa integrasi basis data eksternal mampu mengeliminasi risiko halusinasi secara signifikan pada domain. Oleh karena itu, penelitian ini menggunakan metode RAG ke dalam sistem wawancara kerja. Penggunaan RAG dalam penelitian ini tidak hanya berfungsi untuk mengotomatisasi pembuatan pertanyaan, tetapi juga menjadi metode utama untuk memvalidasi kesesuaian fakta antara pertanyaan yang dihasilkan dari data referensi yang disimpan 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 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 menggunakan pendekatan kuantitatif dengan metode eksperimental untuk membangun sistem wawancara cerdas. Fokus utama penelitian adalah implementasi metode *Retrieval-Augmented Generation (RAG)* sebagai metode inti pemrosesan dalam sistem wawancara kerja otomatis. Penerapan metode RAG ini digunakan untuk mengoptimalkan kinerja *Large Language Model (LLM)* pada model LLaMA-3.3-70B, sehingga mampu menghasilkan pertanyaan yang valid dan relevan berdasarkan dokumen pelamar.

Dalam mekanisme ini, RAG bertugas mengambil informasi kontekstual yang faktual dari basis data *vektor* berdasarkan hasil transkripsi jawaban lisan pelamar sebagai konteks pendukung sebelum diproses oleh LLM. Interaksi sistem dirancang menggunakan pendekatan *multimodal audio-visual*, input suara pengguna menggunakan teknologi *Speech-to-Text* dari model *whisper*, sementara response sistem disampaikan melalui *edge text-to-speech*. Tujuan utama penelitian ini adalah membangun sistem yang mampu mengotomatisasi pembuatan pertanyaan wawancara secara *real-time* dengan mengurangi tingkat halusinasi melalui verifikasi dokumen pelamar.



Gambar 3.1 Rancangan Penelitian



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

Pelaksanaan penelitian sistem ini disusun secara sistematis melalui serangkaian langkah yang terstruktur. Seluruh tahapan penelitian mengacu pada kerangka kerja yang divisualisasikan dalam Gambar 3.1.

1. Identifikasi Masalah & Studi Literatur

Penelitian ini diawali dengan tahap identifikasi kebutuhan dan studi literatur untuk menetapkan spesifikasi sistem yang akan dibangun. Pada tahap ini, perumusan masalah difokuskan pada kebutuhan otomatisasi wawancara kerja. untuk menjawab kebutuhan tersebut, dilakukan pencarian referensi mengenai metode *Retrieval-Augmented Generation* (RAG) dengan LLAMA-3-70B, serta penggunaan fitur suara melalui *whisper* dan *Edge Text-To-Speech*.

2. Pengumpulan & Preprocessing Data

Setelah kebutuhan sistem diidentifikasi, tahap selanjutnya adalah pengumpulan dan pra-pemrosesan data. Penelitian ini menggunakan data sekunder yang diperoleh dari platform Kaggle, yaitu dataset *HR Interview Questions and Ideal answers* serta *Software Engineering Interview Questions Dataset*. Kedua dataset tersebut kemudian melalui tahap pra-pemrosesan yang diawali dengan pembersihan data, lalu dilanjutkan dengan konversi teks menjadi representasi numerik (*vector embedding*) menggunakan model *transformer*.

Kumpulan *vektor* ini kemudian disimpan ke dalam basis data *vektor* (ChromaDB) agar dapat beroperasi sebagai basis pengetahuan (*knowledge base*) utama bagi sistem. Nantinya, untuk menyusun kueri pencarian yang dinamis, sistem akan menggunakan parameter posisi pekerjaan dan tingkat kesulitan, serta mengekstrak kata kunci teknologi langsung dari jawaban pengenalan lisan pengguna sebagai konteks sistem.

3. Perancangan RAG & Prompt Engineering

Tahap ini berfokus pada perancangan mekanisme *retriever* yang bertugas mencari dan menarik materi pertanyaan wawancara yang relevan dari basis pengetahuan (*vector store*). Proses pencarian dinamis ini dihitung berdasarkan metrik kedekatan jarak *vektor* menggunakan algoritma bawaan dari ChromaDB yaitu perhitungan *Squared L2 Distance*. Setelah konteks yang tepat ditemukan, tahapan dilanjutkan



© 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

dengan perancangan prompt engineering untuk menyusun instruksi spesifik bagi model LLaMA-3.3-70B. Prompt ini dirancang untuk memosisikan AI sebagai pewawancara teknis yang profesional, sekaligus memberikan batasan tegas agar AI hanya mengajukan pertanyaan berdasarkan fakta keahlian yang diekstrak dari jawaban pengenalan kandidat di awal sesi wawancara.

4. Evaluasi Kinerja RAG

Proses selanjutnya adalah pengujian terhadap evaluasi kinerja RAG, yang berfokus mengukur kualitas output yang dihasilkan. Penelitian ini menggunakan kerangka kerja RAGAs untuk mengevaluasi kinerja metode RAG. Tujuan utamanya adalah untuk memastikan bahwa pertanyaan wawancara yang dibuat oleh sistem benar-benar relevan dan selaras dengan konteks riwayat keahlian yang diucapkan oleh pelamar pada sesi pengenalan, sehingga meminimalisir terjadinya halusinasi pada kecerdasan buatan.

5. Implementasi Sistem ke Aplikasi Web

Tahapan ini bertujuan untuk mengubah rancangan model RAG menjadi perangkat lunak berbasis web yang fungsional. Proses ini diawali dengan perancangan antarmuka pengguna sebagai media interaksi, selanjutnya, dilakukan penulisan kode program untuk mengimplementasikan logika RAG menjadi layanan backend yang siap memproses data. Tahapan ini diakhiri dengan integrasi sistem, yaitu menghubungkan model kecerdasan buatan dengan antarmuka web melalui mekanisme *Application Programming Interface (API)*.

6. Pengujian Aplikasi & Analisis Hasil

Tahap selanjutnya, mencakup menyeluruh terhadap aplikasi dan dokumentasi hasil. Sebelum laporan dibuat, pengujian sistem dilakukan menggunakan tiga metode, black box testing untuk memastikan bahwa fitur berjalan lancar dan pengujian SUS untuk mengukur tingkat kepuasan pengguna serta NPS untuk mengukur tingkat loyalitas pengguna.

7. Pembuatan Laporan Akhir

Setelah rangkaian pengujian selesai, penelitian ditutup dengan tahap pembuatan laporan akhir. Dokumen ini bertujuan untuk mendokumentasikan seluruh proses penelitian, mulai dari identifikasi kebutuhan, pengumpulan, pengolahan data,



© Hak Cipta milik Politeknik Negeri Jakarta

perancang arsitektur RAG, implementasi sistem, integrasi, dan pengujian sistem yang dilakukan.

3.3 Objek Penelitian

Penelitian ini berfokus pada pengembangan sistem wawancara kerja otomatis yang menerapkan arsitektur *Retrieval-Augmented Generation* (RAG) menggunakan model LLaMA-3.3-70B. Secara spesifik, penelitian ini difokuskan pada kemampuan sistem dalam memproses konteks keahlian dari jawaban pengenalan pengguna secara langsung untuk menghasilkan pertanyaan wawancara teknis yang dinamis dan relevan dengan posisi pekerjaan yang dilamar.

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



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 Analisis Kebutuhan

Berdasarkan studi literatur dari berbagai jurnal yang telah dilakukan, beberapa kebutuhan berhasil diidentifikasi untuk penelitian ini. Kebutuhan tersebut mencakup kebutuhan dataset, kebutuhan fungsional, dan kebutuhan non-fungsional. semua kebutuhan tersebut perlu dilengkapi untuk menjamin kelancaran penelitian.

4.1.1 Kebutuhan Dataset

Dalam sistem berbasis *Retrieval-Augmented Generation* (RAG), dataset memiliki peran penting karena berfungsi sebagai sumber pengetahuan yang digunakan oleh sistem untuk memberikan respons yang relevan dan kontekstual. Dataset yang digunakan harus memiliki struktur data yang jelas serta berisi informasi yang berkaitan dengan topik wawancara kerja. Pada penelitian ini, dataset yang digunakan berasal dari dua sumber yang tersedia pada platform Kaggle. Berikut ini dataset yang akan digunakan pada Tabel 2.

Tabel 2 Kebutuhan Dataset

Kode	Nama	Deksripsi
KD-1	HR Interview Questions and Ideal answers	Dataset ini berisi kumpulan pertanyaan wawancara yang umum diajukan oleh HR beserta contoh jawaban yang dianggap ideal. Dataset ini digunakan sebagai sumber pengetahuan untuk menghasilkan pertanyaan wawancara non-teknis serta membantu sistem dalam menganalisis jawaban pengguna selama simulasi wawancara.



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

KD-2	Software Engineering Interview Questions Dataset	Dataset ini berisi berbagai pertanyaan wawancara teknis yang berkaitan dengan bidang rekayasa perangkat lunak, seperti konsep pemrograman, algoritma, struktur data, serta praktik terbaik dalam pengembangan perangkat lunak. Dataset ini digunakan untuk memperkaya <i>knowledge base</i> sistem dalam menghasilkan pertanyaan wawancara teknis.
------	--	--

4.1.2 Kebutuhan Perangkat Keras

Platform google colab digunakan untuk melakukan rancangan RAG, Pengujian RAG. Spesifikasi perangkat keras untuk pengembangan model dapat dilihat pada Tabel 3.

Tabel 3 Kebutuhan Perangkat Keras Pengembangan Model

Kode	Nama	Deksripsi
KPKM-1	GPU	Nvidia Tesla T4
KPKM-2	CPU	Intel(R) Xeon(R) CPU @ 2.20GHz
KPKM-3	RAM	Minimal 16 GB

Berbeda dengan spesifikasi pengembangan model, kebutuhan spesifikasi komputer untuk pengembangan web memiliki spesifikasi yang jauh lebih rendah yang tercantum pada Tabel 4.

Tabel 4 Kebutuhan Perangkat Keras Pengembangan Web

Kode	Nama	Deksripsi
KPKW-1	CPU	Minimal 4 Core
KPKW-2	RAM	Minimal 8 GB

Kebutuhan pengembangan web lebih rendah karena tidak membutuhkan komputasi intensif, sehingga dapat menggunakan laptop pribadi.

4.1.3 Kebutuhan Perangkat Lunak

Dalam pengembangan sistem RAG, diperlukan perangkat lunak untuk mendukung proses RAG, pengolahan data, Pengembangan antarmuka sistem, serta implementasi dan pengujian sistem. Selain itu, dibutuhkan pula berbagai library dan framework pendukung yang digunakan dalam proses pengembangan, mulai dari pengelolaan model bahasa, pemrosesan vektor, hingga integrasi antarkomponen



© 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

sistem. Kebutuhan perangkat lunak dan library yang digunakan dapat dilihat pada Tabel 5.

Tabel 5 Kebutuhan Perangkat Lunak Pengembangan Model

Kode	Nama	Deksripsi
KPLMM-1	Python	Bahasa Pemrograman dalam pengembangan RAG dan backend sistem.
KPLMM-2	Google Colab	Digunakan untuk proses pembersihan data, pengembangan dan evaluasi
KPLMM-3	Groq	Layanan cloud yang menyediakan akses berbagai model AI, termasuk model bahasa LLaMA-3.3-70B yang digunakan sebagai reasoning engine (menyusun pertanyaan wawancara dan narasi feedback/laporan, bukan untuk penilaian) dalam pengembangan RAG, serta Whisper-Large-V3 untuk speech-to-text.
KPLMM-4	ChromaDB	Vector database lokal yang digunakan untuk menyimpan, mengindeks, dan memanggil data referensi wawancara (knowledge base) berdasarkan kedekatan jarak vektor.
KPLMM-5	Sentence-Transformers	Framework Python yang digunakan untuk memuat model <i>paraphrase-multilingual-MiniLM-L12-v2</i> untuk melakukan proses tokenisasi dan ekstraksi teks menjadi angka vektor.
KPLMM-6	RAGAS (Retrieval-Augmented Generation Assessment)	Framework evaluasi otomatis yang digunakan untuk mengukur kualitas sistem RAG berdasarkan metrik Context Precision, Faithfulness, dan Answer Relevance.
KPLMM-5	VPS	Digunakan untuk deployment backend
KPLMM-6	PostgreSQL	Digunakan untuk menyimpan data
KPLMM-7	Pandas	Digunakan untuk manipulasi dan analisis data
KPLMM-8	Numpy	Digunakan untuk komputasi numerik
KPLMM-9	Scikit-Learn	Digunakan untuk mengekstrak kata kunci dari teks referensi
KPLMM-10	Re (Regular Expression)	Digunakan untuk memecahkan paragraf panjang menjadi potongan-potongan kalimat tunggal
KPLMM-11	Sastrawi	Digunakan untuk proses stemming Bahasa Indonesia, yaitu mengubah kata berimbuhan menjadi bentuk kata dasarnya.



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

Tahap selanjutnya adalah identifikasi perangkat lunak yang digunakan untuk membangun antarmuka sistem berbasis web. Rincian dari keseluruhan kebutuhan tersebut dilihat pada Tabel 6.

Tabel 6 Kebutuhan Perangkat Lunak Pengembangan Web

Kode	Nama	Deksripsi
KPLPW-1	React.js	Library JavaScript berbasis komponen yang digunakan untuk membangun antarmuka aplikasi wawancara kerja secara reaktif dan dinamis.
KPLPW-2	Tailwind.css	Framework utility-first CSS yang digunakan untuk mempercepat proses perancangan desain antarmuka web yang responsif, modern, dan konsisten.
KPLPW-3	Vercel	Digunakan untuk deployment aplikasi antarmuka berbasis web
KPLPW-4	Web Speech API / Recorder.js	Library/API browser yang digunakan untuk menangkap input audio dari mikrofon kandidat secara langsung pada halaman web sebelum dikirim ke backend.

4.1.4 Kebutuhan Fungsional

Analisis kebutuhan pada penelitian ini ditujukan sebagai implementasi sistem wawancara kerja tanpa menghilangkan fitur-fitur penting seperti autentikasi dan penyimpanan data. Fitur utama aplikasi dapat dilihat pada Tabel 7.

Tabel 7 Kebutuhan Fungsional

Kode	Nama	Deskripsi
KF-1	Daftar	Pengguna dapat mendaftarkan akun ke dalam sistem
KF-2	Masuk	Pengguna dapat masuk ke dalam sistem menggunakan akun yang telah dibuat
KF-3	Memilih Posisi Wawancara	Sistem memungkinkan pengguna memilih kategori, <i>role</i> , dan level posisi pekerjaan
KF-4	Memulai Sesi Wawancara	Sistem memungkinkan pengguna memulai simulasi wawancara dengan AI interviewer setelah memilih posisi yang diinginkan.
KF-7	Merekam Jawaban Kandidat	Sistem menerima dan merekam jawaban yang diberikan oleh pengguna selama sesi wawancara berlangsung.



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-8	Menganalisis Jawaban Kandidat	Sistem melakukan analisis terhadap jawaban pengguna menggunakan AI untuk menilai kualitas jawaban.
KF-9	Menyimpan Riwayat Wawancara	Sistem menyimpan data sesi wawancara yang telah dilakukan oleh pengguna.

4.1.5 Kebutuhan Non-Fungsional

Selain kebutuhan fungsional, sistem juga harus memenuhi beberapa kebutuhan non-fungsional berikut untuk meningkatkan kenyamanan penggunaan pada Tabel 8.

Tabel 8 Kebutuhan Non-Fungsional

Kode	Nama	Deksripsi
KNF-1	Performa Sistem	Sistem harus mampu menghasilkan pertanyaan wawancara secara otomatis dalam waktu yang cepat sehingga percakapan dengan AI dapat berlangsung secara <i>real-time</i> .
KNF-2	Kecepatan Analisis	Sistem harus mampu melakukan analisis terhadap jawaban kandidat dan menghasilkan evaluasi dalam waktu yang singkat setelah sesi wawancara selesai.
KNF-3	Keamanan Data	Sistem harus menjaga keamanan data pengguna, termasuk data wawancara, rekaman suara, dan hasil analisis agar tidak dapat diakses oleh pihak yang tidak berwenang.

4.2 Perancangan Sistem

Tahap perancangan sistem dilakukan setelah proses analisis kebutuhan selesai. Pada tahap, sistem mulai dengan memetakan struktur serta alur kerja yang akan digunakan sebelum tahap implementasi atau pembangunan sistem dilakukan. Tujuannya adalah agar sistem yang dikembangkan benar-benar sesuai dengan kebutuhan pengguna serta tujuan yang ingin dicapai.

Dalam proses perancangan ini digunakan pendekatan *Unified Modeling Language* (UML) untuk membantu menggambarkan alur dan struktur sistem secara lebih jelas. Beberapa diagram yang digunakan antara lain:

1. *Use Case Diagram*, untuk menggambarkan interaksi antara pengguna dengan sistem yang dibangun.



Hak Cipta :

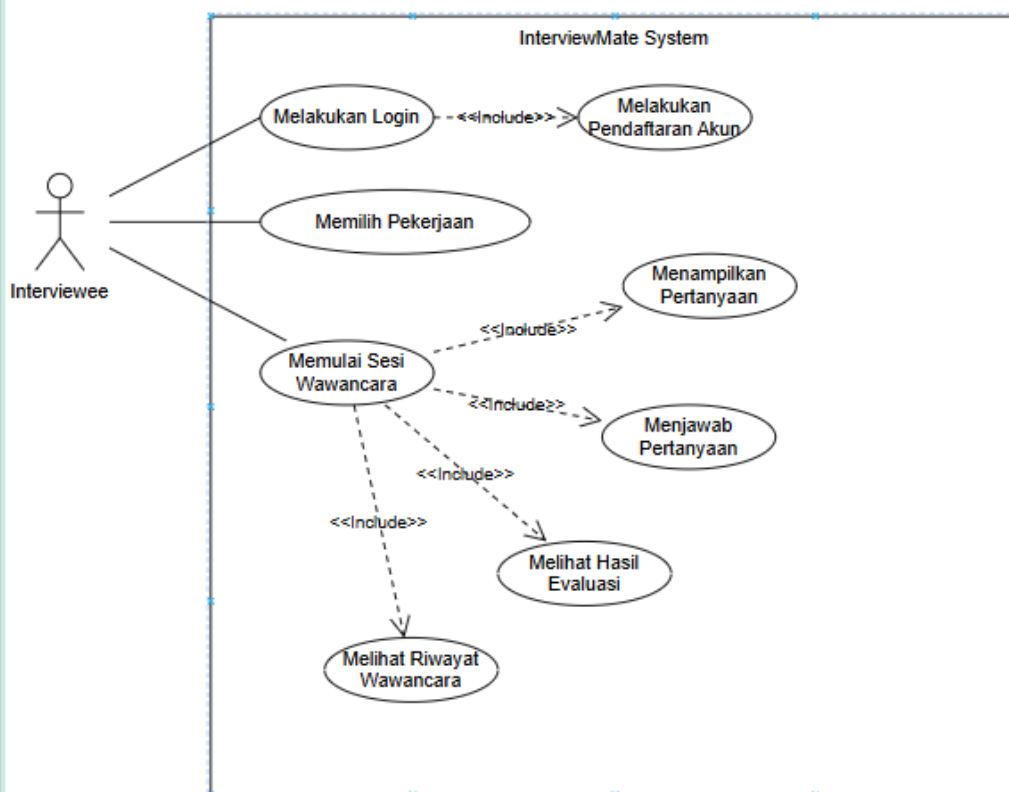
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan 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. *Activity Diagram*, untuk mendeskripsikan alur proses utama dalam sistem secara logis dan berurutan.
3. *Sequence Diagram*, untuk menggambarkan interaksi antar objek secara lebih rinci dalam suatu sistem.
4. *Class Diagram*, untuk menjelaskan rancangan basis data yang digunakan dalam menyimpan informasi.

Melalui keempat diagram tersebut, perancangan sistem dapat memberikan gambaran yang lebih menyeluruh mengenai fungsi sistem, alur proses yang terjadi, serta struktur data yang diperlukan, sehingga membantu dalam membangun sistem yang lebih terstruktur dan efektif.

4.2.1 Use Case Diagram

Use case diagram pada penelitian ini dikembangkan berdasarkan kebutuhan fungsional yang telah diidentifikasi sebelumnya. Diagram ini menggambarkan interaksi antara pengguna dengan sistem serta menunjukkan berbagai fungsi yang dapat dijalankan oleh sistem. *Use case diagram* dari sistem yang dikembangkan dapat dilihat pada Gambar dibawah ini



Gambar 4.1 Use Case Diagram



© 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

Use case diagram pada Gambar 4.1 menggambarkan sistem wawancara kerja berbasis AI yang menunjukkan interaksi antara pengguna (*interviewee*) dengan sistem. Diagram ini menjelaskan berbagai aktivitas yang dapat dilakukan pengguna saat menggunakan aplikasi *InterviewMate*. Untuk dapat mengakses sistem, pengguna berinteraksi melalui *use case* Melakukan Login, yang pada rancangan ini alurnya juga terhubung dengan proses pembuatan akun baru melalui *use case* Melakukan Pendaftaran Akun. Setelah mendapatkan akses, pengguna dapat memulai persiapan dengan memilih parameter wawancara melalui *use case* Memilih Pekerjaan, di mana pengguna menentukan kategori, *role*, dan level posisi yang diinginkan.

Selanjutnya, pengguna dapat menginisiasi proses wawancara melalui *use case* Memulai Sesi Wawancara. Di dalam alur sesi ini, terdapat beberapa interaksi yang wajib dilalui oleh pengguna. Pertama, pengguna akan berhadapan dengan antarmuka yang memunculkan soal melalui *use case* Menampilkan Pertanyaan, kemudian pengguna memberikan responsnya secara langsung melalui *use case* Menjawab Pertanyaan. Segala proses kompleks di belakang layar seperti transkripsi suara menjadi teks dan analisis kecerdasan buatan akan dieksekusi oleh sistem selama proses ini berlangsung.

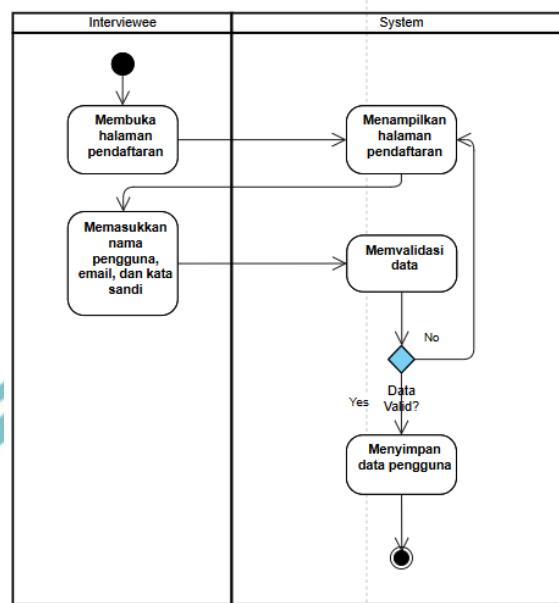
Setelah rangkaian tanya jawab selesai dan sesi diakhiri, bagian akhir dari sesi ini akan mengarahkan pengguna untuk meninjau performa dan skor mereka melalui *use case* Melihat Hasil Evaluasi. Di luar rangkaian wawancara aktif tersebut, pengguna juga memiliki akses terpisah untuk meninjau kembali seluruh rekam jejak sesi wawancara yang telah berlalu melalui *use case* Melihat Riwayat Wawancara.

4.2.2 Activity Diagram

Activity diagram digunakan untuk menggambarkan alur aktivitas yang terjadi dalam sistem secara lebih rinci dan terstruktur. Diagram ini menunjukkan langkah-langkah proses yang dilakukan oleh pengguna dan sistem, mulai dari proses awal hingga proses akhir.



1. Proses Pendaftaran Akun



Gambar 4.2 Activity Diagram Proses Pendaftaran Akun

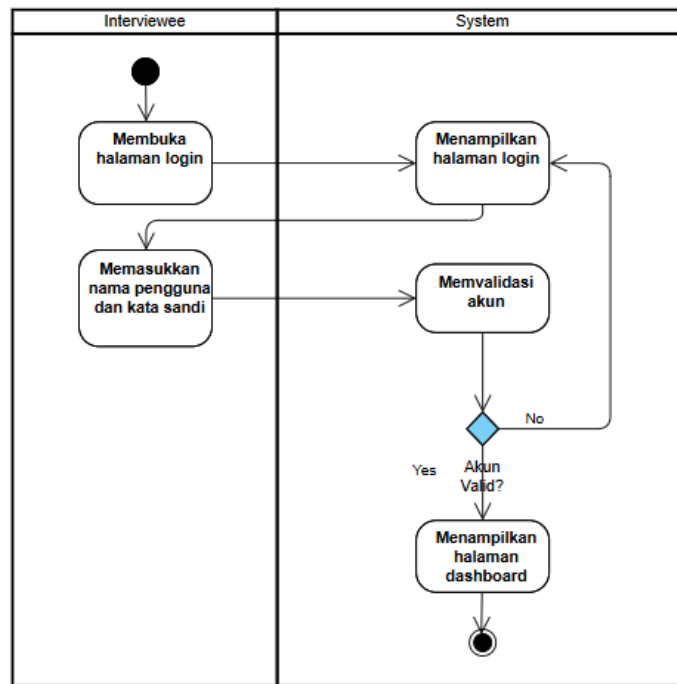
Activity diagram pada Gambar 4.2 merupakan alur pendaftaran akun bagi pengguna. Interaksi ini dimulai saat pengguna mengakses fitur pendaftaran, di mana sistem akan merespons dengan memunculkan halaman formulir pendaftaran. Pengguna kemudian melengkapi informasi yang dibutuhkan dengan memasukkan nama pengguna, alamat email, dan kata sandi. Setelah data diserahkan, sistem mengambil alih untuk melakukan proses validasi guna memastikan kelengkapan dan kebenaran format input. Apabila data terdeteksi tidak valid, sistem akan menahan proses dan mengarahkan pengguna kembali ke halaman pendaftaran untuk melakukan perbaikan. Namun, jika seluruh data telah dinyatakan valid, sistem akan menyimpan data pengguna tersebut ke dalam database, dan menyelesaikan proses dengan mengarahkan pengguna menuju halaman login untuk mengakses akun baru mereka.

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



2. Proses Login Akun



Gambar 4.3 Activity Diagram Proses Login Akun

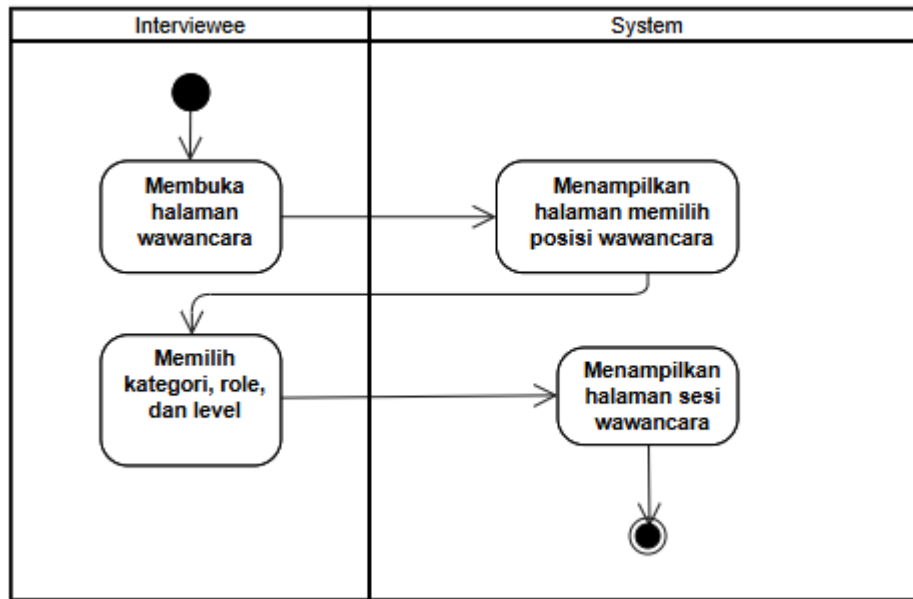
Activity diagram pada Gambar 4.3 merupakan alur proses otentikasi atau login yang dilakukan oleh pengguna. Alur dimulai ketika pengguna mengakses sistem, yang kemudian direspons oleh sistem dengan menampilkan halaman login. Selanjutnya, pengguna memasukkan kredensial mereka yang terdiri dari nama pengguna dan kata sandi. Setelah data dimasukkan, sistem mengambil alih proses untuk melakukan validasi akun. Pada tahap pengecekan ini, apabila kredensial yang diberikan tidak valid, sistem akan mengarahkan pengguna kembali ke halaman login untuk mencoba lagi. Namun, jika akun dinyatakan valid, sistem akan memberikan akses masuk dan mengarahkan pengguna ke halaman dashboard utama.

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. Proses Memilih Pekerjaan



Gambar 4.4 Activity Diagram Proses Memilih Pekerjaan

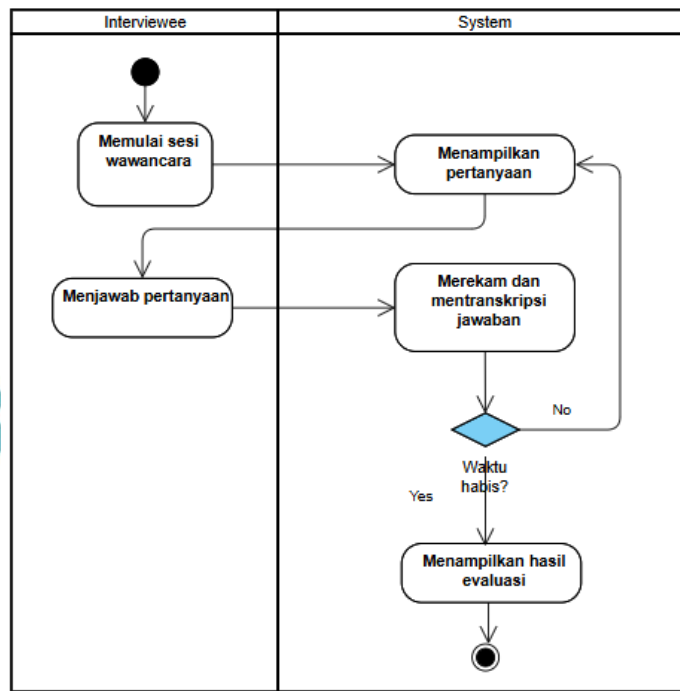
Activity diagram pada Gambar 4.4 merupakan alur persiapan pengguna sebelum memulai simulasi wawancara. Proses ini diinisiasi ketika pengguna mengakses menu untuk membuka halaman wawancara, yang kemudian direspons oleh sistem dengan menampilkan antarmuka pemilihan posisi wawancara. Pada halaman tersebut, pengguna diminta untuk menentukan spesifikasi pekerjaan yang ingin dilatih dengan memilih Kategori, Role, dan Level keahlian melalui opsi dropdown yang tersedia. Setelah pengguna selesai memilih dan mengirimkan preferensi tersebut, sistem akan langsung memproses pilihan itu dan beralih menampilkan halaman sesi wawancara utama kepada pengguna untuk memulai tahapan wawancara.

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. Proses Sesi Wawancara



Gambar 4.5 Activity Diagram Proses Sesi Wawancara

Activity diagram pada Gambar 4.5 merupakan alur pelaksanaan simulasi wawancara. Proses dimulai ketika pengguna memasuki sesi wawancara, yang memicu sistem untuk secara otomatis menghasilkan dan menampilkan pertanyaan wawancara melalui *Ima AI* (InterviewMate AI). Pengguna kemudian merespons dengan menjawab pertanyaan tersebut secara lisan. Sistem akan langsung menangkap respons tersebut, merekamnya, dan menampilkan transkripsi teksnya secara *real-time* di layar.

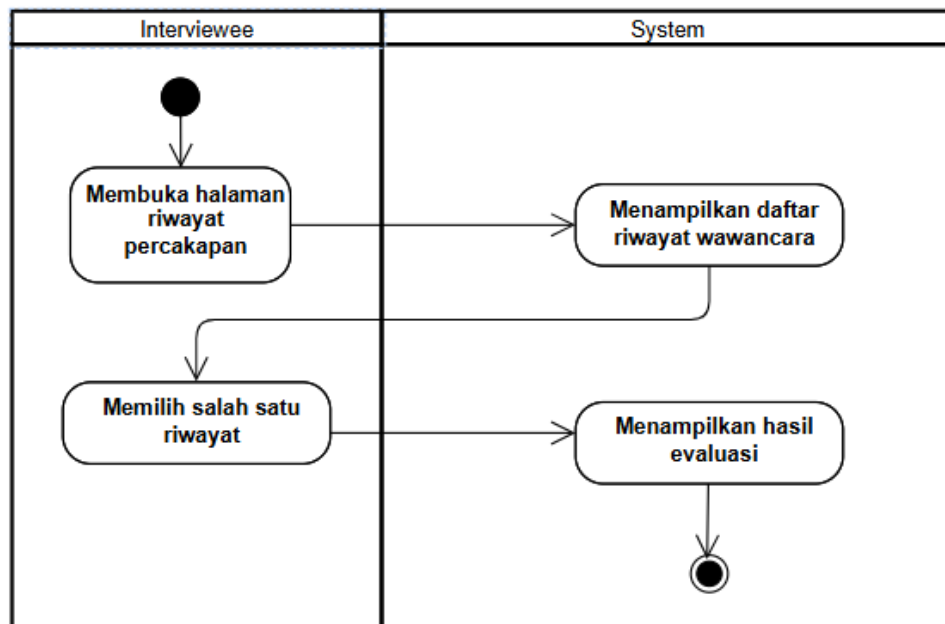
Setelah satu siklus tanya jawab selesai, sistem akan melakukan pengecekan terhadap durasi wawancara. Apabila batasan waktu belum habis, sistem akan melakukan perulangan dengan kembali menghasilkan dan melontarkan pertanyaan selanjutnya. Sebaliknya, jika durasi telah habis atau pengguna memutuskan untuk mengakhiri sesi lebih awal melalui tombol Akhiri Wawancara sistem akan menghentikan sesi tersebut secara otomatis dan langsung beralih memproses serta menampilkan halaman hasil evaluasi keseluruhan.

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



5. Melihat Riwayat Wawancara



Gambar 4.6 Activity Diagram Melihat Riwayat Wawancara

Activity diagram pada Gambar 4.6 mendeskripsikan alur interaksi saat pengguna meninjau kembali rekam jejak sesi wawancara yang telah diselesaikan. Proses ini diinisiasi ketika pengguna membuka menu Interview History, yang memicu sistem bekerja di latar belakang untuk melakukan kueri pengambilan data dari database. Setelah data berhasil dimuat, sistem menyajikan daftar riwayat sesi wawancara tersebut di layar. Pengguna kemudian dapat memilih salah satu riwayat spesifik yang ingin dianalisis lebih lanjut. Sebagai respons akhir, sistem akan menampilkan halaman hasil evaluasi secara komprehensif. Halaman ini memuat metrik penilaian AI berupa laporan akhir yang mencakup rangkuman performa, kekuatan teknis, area yang perlu ditingkatkan, serta rekomendasi tindak lanjut, yang juga dilengkapi dengan lampiran transkrip percakapan utuh beserta feedback spesifik pada setiap poin jawaban kandidat.

4.2.3 Sequence Diagram

Sequence diagram merupakan salah satu diagram pada UML (*Unified Modeling Language*) yang digunakan untuk menggambarkan interaksi antar objek secara terperinci. Diagram ini memodelkan aliran logika dalam sebuah sistem dengan menyoroti pertukaran pesan antar objek berdasarkan urutan waktu kejadian. Tujuan

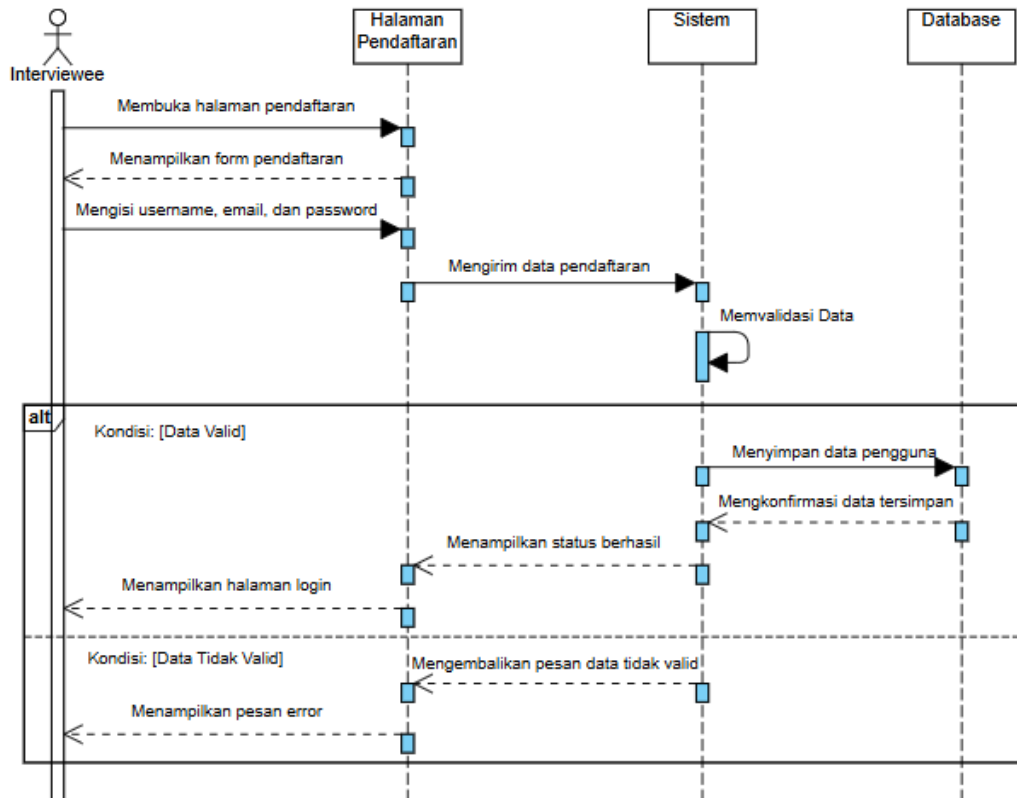


Hak Cipta :

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

utamanya adalah untuk memperlihatkan bagaimana komponen-komponen sistem saling berkomunikasi dari awal hingga akhir suatu proses atau use case tertentu.

1. Proses Pendaftaran Akun



Gambar 4.7 Sequence Diagram Proses Pendaftaran Akun

Sequence diagram pada Gambar 4.7 menggambarkan urutan proses pendaftaran akun oleh *Interviewee*. Proses diinisiasi saat pengguna membuka halaman pendaftaran, yang direspons oleh sistem dengan menampilkan form pendaftaran. Setelah pengguna mengisi username, email, dan password, antarmuka akan mengirim data pendaftaran tersebut ke Sistem. Sistem kemudian melakukan *self-message* untuk memvalidasi kelayakan data secara internal.

Diagram ini menggunakan *Combine Fragment Alternative* (alt) untuk merepresentasikan percabangan logika berdasarkan hasil validasi. Pada kondisi Data Valid, Sistem akan meneruskan proses dengan menyimpan data pengguna ke Database. Setelah Database mengkonfirmasi data tersimpan, Sistem akan menampilkan status berhasil ke antarmuka, yang kemudian merespons pengguna dengan menampilkan halaman login. Sebaliknya, pada kondisi Data Tidak Valid, Sistem menolak penyimpanan dan mengembalikan pesan data tidak valid ke

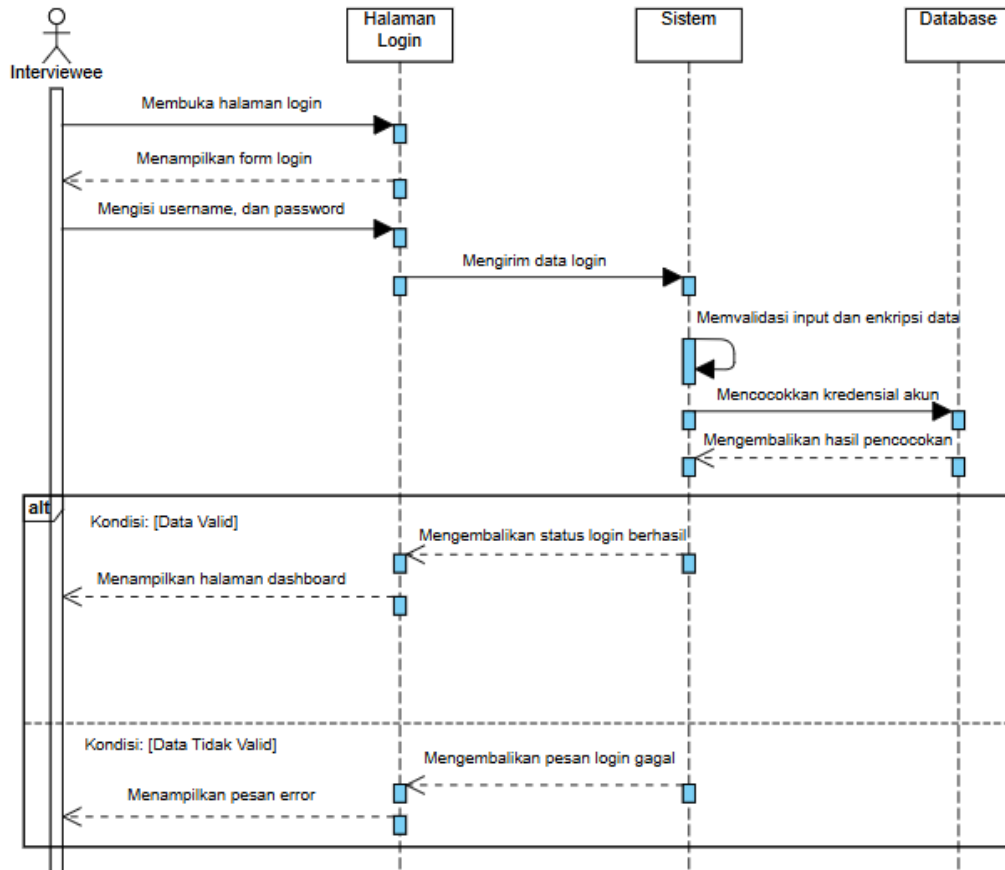


Hak Cipta :

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

antarmuka, yang berujung pada menampilkan pesan *error* kepada pengguna agar dapat memperbaiki isian datanya.

2. Proses Login Akun



Gambar 4.8 Sequence Diagram Proses Login Akun

Sequence diagram pada Gambar 4.8 menguraikan alur interaksi antar objek selama proses autentikasi atau login ke dalam sistem. Proses ini diawali ketika *Interviewee* mengakses halaman login, yang kemudian direspons oleh antarmuka dengan menampilkan form login di layar. Setelah pengguna mengisi username dan password lalu menekan tombol kirim, data kredensial tersebut diteruskan ke Sistem. Sebelum melakukan pencocokan data, Sistem terlebih dahulu melakukan *self-message* untuk memvalidasi input dan mengenkripsi data sandi demi keamanan. Setelah tahap validasi awal tersebut selesai, Sistem akan berinteraksi dengan Database untuk mencocokkan kredensial akun. Database kemudian memproses kueri tersebut dan mengembalikan hasil pencocokan kembali ke Sistem.

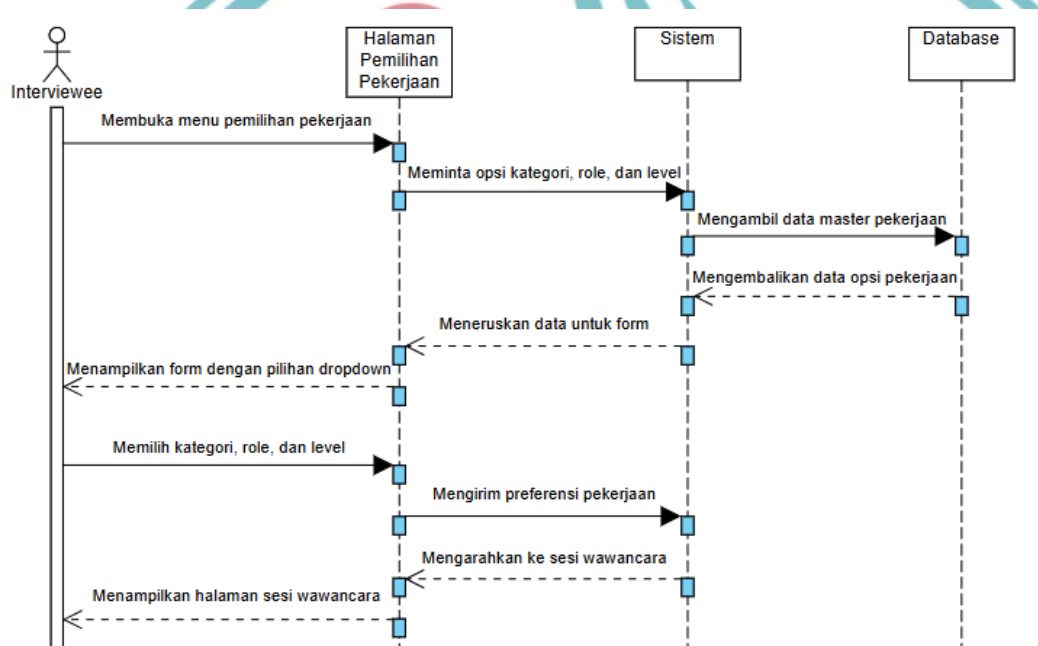


Hak Cipta :

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 mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Diagram ini mengimplementasikan *Combine Fragment Alternative* (alt) untuk mengelola dua kemungkinan hasil autentikasi. Pada kondisi [Akun Valid], yang berarti kredensial pengguna cocok dengan data yang tersimpan, Sistem akan memberikan konfirmasi status berhasil ke antarmuka, yang kemudian mengarahkan pengguna untuk masuk ke halaman utama aplikasi. Sebaliknya, pada kondisi [Akun Tidak Valid], Sistem mendeteksi adanya ketidakcocokan data dan mengembalikan pesan gagal, sehingga antarmuka akan menampilkan pesan *error* kepada pengguna agar mereka dapat memeriksa kembali data yang dimasukkan.

3. Proses Memilih Pekerjaan



Gambar 4.9 Sequence Diagram Proses Memilih Pekerjaan

Sequence diagram pada Gambar 4.9 mengilustrasikan alur sistem pada tahap persiapan sebelum simulasi wawancara dimulai. Interaksi diawali ketika pengguna mengakses menu pemilihan pekerjaan. Sebelum antarmuka menampilkannya kepada pengguna, Sistem akan melakukan kueri ke Database untuk mengambil data master yang berisi opsi kategori, *role*, dan level pekerjaan. Setelah data referensi tersebut dikembalikan, antarmuka akan menampilkan formulir dengan opsi dropdown yang dinamis.

Selanjutnya, pengguna menentukan spesifikasi posisi yang diinginkan dengan memilih opsi pada formulir tersebut. Setelah pengguna mengirimkan preferensi pekerjaannya, antarmuka akan meneruskan data tersebut ke Sistem untuk diproses.

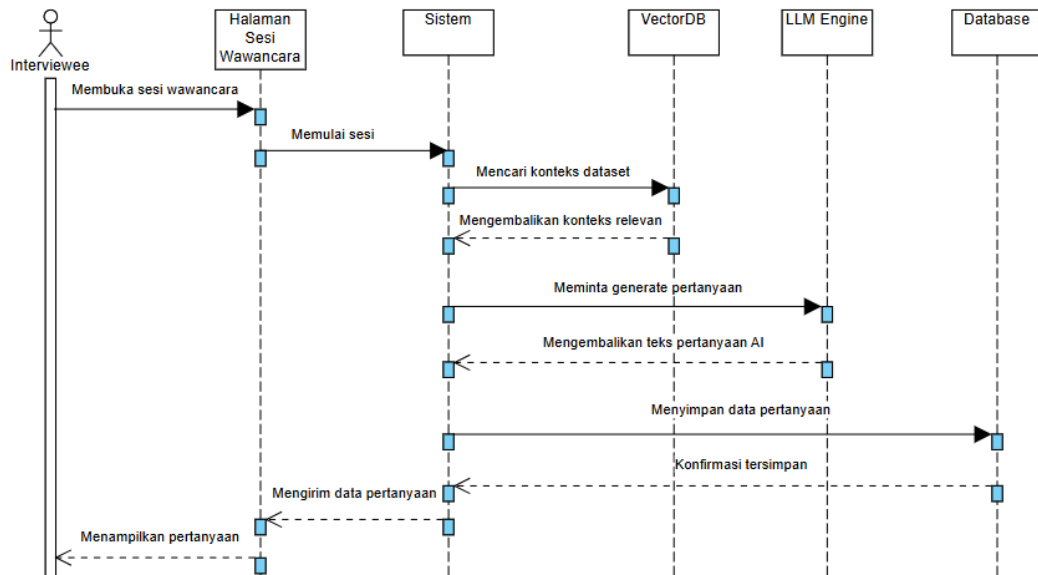


Hak Cipta :

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

Sebagai hasil akhir dari alur ini, Sistem memvalidasi kesiapan sesi dan menginstruksikan antarmuka untuk langsung mengarahkan dan menampilkan halaman sesi wawancara utama kepada pengguna.

4. Sequence Diagram Proses Menampilkan Pertanyaan



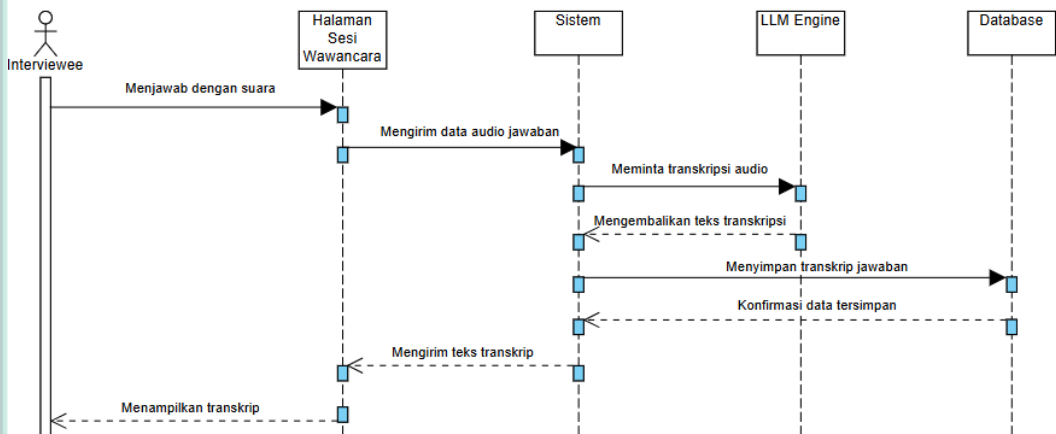
Gambar 4.10 Sequence Diagram Proses Menampilkan Pertanyaan

Sequence diagram pada Gambar 4.10 menguraikan alur sistem dalam menghasilkan pertanyaan wawancara menggunakan pendekatan *Retrieval-Augmented Generation* (RAG). Proses ini diinisiasi ketika antarmuka Halaman Wawancara mengirimkan permintaan pertanyaan baru kepada Sistem. Sistem kemudian bertindak dengan melakukan pencarian (*retrieval*) ke dalam VectorDB untuk mengekstrak dokumen atau konteks dataset yang relevan dengan spesifikasi posisi kandidat.

Setelah konteks yang tepat diperoleh, Sistem menggabungkannya dengan prompt instruksi dan mengirimkannya ke LLM Engine. LLM Engine kemudian memproses data tersebut untuk merumuskan (*generate*) pertanyaan wawancara yang spesifik dan kontekstual. Teks pertanyaan yang dihasilkan kemudian dikembalikan ke Sistem, untuk selanjutnya disimpan secara permanen ke dalam Database utama sebagai riwayat sesi. Setelah proses penyimpanan dikonfirmasi, Sistem meneruskan teks pertanyaan tersebut ke Halaman Wawancara untuk ditampilkan secara visual kepada kandidat.



5. Sequence Diagram Proses Menjawab Pertanyaan



Gambar 4.11 Sequence Diagram Proses Menjawab Pertanyaan

Sequence diagram pada Gambar 4.11 menguraikan alur sistem pada tahapan perekaman dan transkripsi jawaban kandidat. Proses dimulai ketika *Interviewee* memberikan respons berupa suara yang direkam melalui antarmuka Halaman Sesi Wawancara. Setelah proses perekaman selesai, antarmuka akan mengirimkan data audio tersebut kepada Sistem.

Selanjutnya, Sistem meneruskan data audio itu ke LLM Engine untuk diproses menggunakan teknologi *Speech-to-Text* agar diubah menjadi bentuk teks. Setelah LLM Engine mengembalikan teks transkripsi yang berhasil diekstrak, Sistem akan langsung menyimpannya ke dalam Database utama sebagai bagian dari riwayat percakapan. Setelah menerima konfirmasi penyimpanan dari Database, Sistem mengirimkan status keberhasilan ke antarmuka, yang kemudian akan memberikan umpan balik visual atau menampilkan teks transkrip tersebut di layar pengguna.

Hak Cipta :

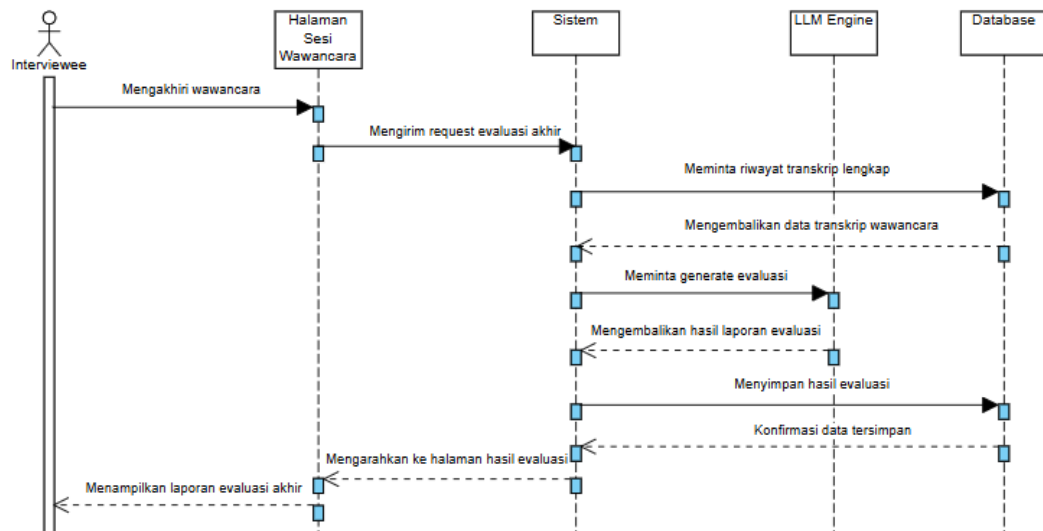
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan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

6. Sequence Diagram Proses Melihat Hasil Evaluasi



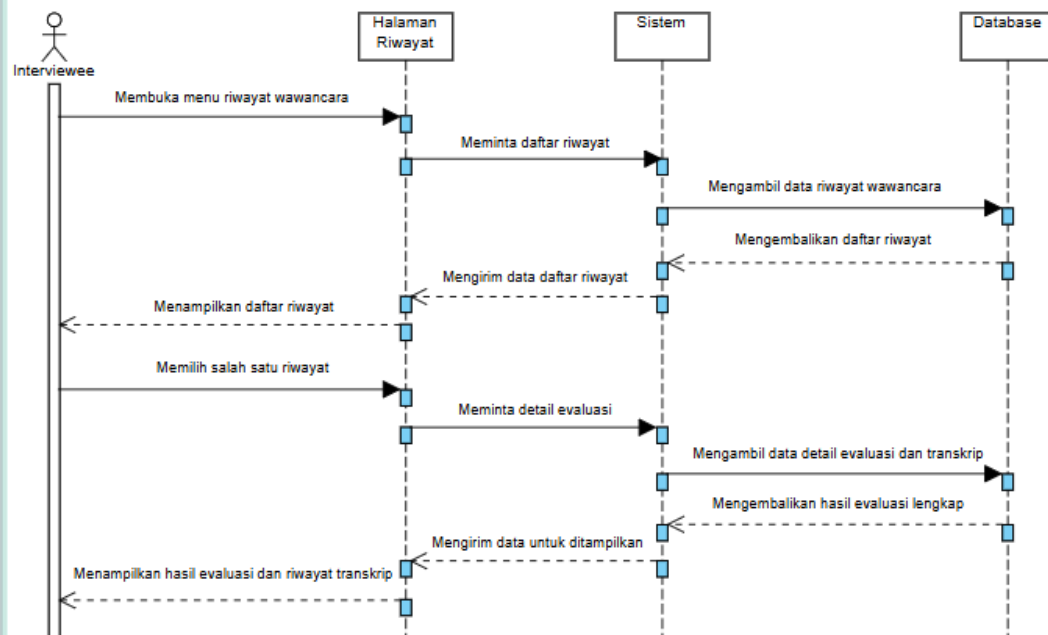
Gambar 4.12 Sequence Diagram Proses Melihat Hasil Evaluasi

Sequence diagram pada Gambar 4.12 menguraikan proses akhir dari simulasi wawancara, yaitu tahap evaluasi dan pembuatan laporan. Alur ini dipicu ketika *Interviewee* memutuskan untuk mengakhiri sesi, atau ketika batas waktu wawancara telah habis, sehingga antarmuka mengirimkan instruksi evaluasi akhir kepada Sistem.

Untuk dapat melakukan penilaian secara komprehensif, Sistem terlebih dahulu melakukan kueri ke Database untuk menarik seluruh riwayat transkrip percakapan selama sesi berlangsung. Seluruh data transkrip tersebut kemudian dikirimkan ke LLM Engine untuk dianalisis. Pada tahap ini, model bahasa bekerja menghasilkan metrik evaluasi yang mencakup skor keseluruhan, analisis kekuatan teknis kandidat, identifikasi kelemahan, serta rekomendasi pembelajaran. Setelah LLM Engine mengembalikan laporan analisis tersebut, Sistem menyimpannya secara permanen ke dalam Database. Sebagai langkah penutup, Sistem merespons antarmuka untuk melakukan peralihan, dan antarmuka menampilkan halaman laporan hasil evaluasi tersebut secara utuh kepada pengguna.



7. Melihat Riwayat Wawancara



Gambar 4.13 Sequence Diagram Melihat Riwayat Wawancara

Sequence diagram pada Gambar 4.13 menjelaskan alur interaksi saat pengguna ingin meninjau kembali rekam jejak simulasi wawancara yang pernah dilakukan. Proses diawali ketika pengguna mengakses menu riwayat melalui antarmuka Halaman Riwayat. Antarmuka kemudian meneruskan permintaan ini kepada Sistem, yang bertugas melakukan kueri ke Database untuk mengambil kumpulan data daftar riwayat wawancara milik pengguna tersebut. Setelah Database mengembalikan data yang diminta, Sistem mengirimkan informasi tersebut ke antarmuka untuk ditampilkan sebagai daftar riwayat di layar.

Selanjutnya, proses memasuki tahap peninjauan detail ketika pengguna memilih salah satu riwayat wawancara spesifik dari daftar tersebut. Antarmuka akan mengirimkan permintaan detail evaluasi kepada Sistem. Sistem kemudian kembali berinteraksi dengan Database untuk menarik data hasil evaluasi lengkap beserta transkrip percakapan pada sesi terkait. Setelah Database mengembalikan data komprehensif tersebut, Sistem meneruskannya kembali ke antarmuka, yang kemudian mengakhiri alur ini dengan menampilkan halaman hasil evaluasi akhir dan riwayat transkrip secara utuh kepada pengguna.

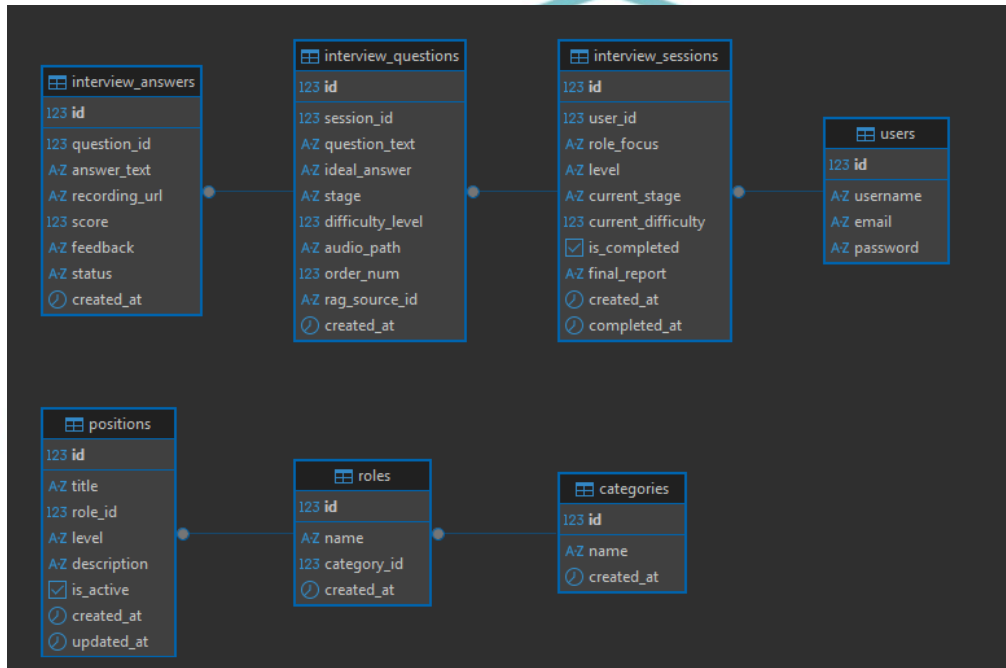


Hak Cipta :

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 mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.3.4 Class Diagram

Class diagram digunakan untuk menggambarkan struktur kelas dalam sistem serta hubungan antar kelas yang berkaitan dengan data yang disimpan. Diagram ini juga menunjukkan atribut dan relasi yang terdapat dalam setiap kelas. Pada penelitian ini, sistem menggunakan database PostgreSQL sebagai media penyimpanan data.



Gambar 4.14 Class Diagram Sistem InterviewMate

Class Diagram pada Gambar 4.14 menunjukkan struktur data yang digunakan dalam sistem InterviewMate beserta hubungan antar entitas yang tersimpan pada database PostgreSQL. Diagram ini menjelaskan tabel-tabel yang digunakan, atribut pada masing-masing tabel, serta keterkaitan antar tabel dalam mendukung proses pada sistem.

Pada sistem ini terdapat beberapa tabel utama yaitu *users*, *categories*, *roles*, *positions*, *interview_sessions*, *interview_questions*, dan *interview_answers*. Tabel *users* digunakan untuk menyimpan data akun pengguna yang dapat mengakses sistem. Selanjutnya tabel *categories* digunakan untuk menyimpan kategori bidang pekerjaan. Tabel *roles* berfungsi untuk menyimpan jenis peran pekerjaan yang terhubung dengan kategori tertentu. Kemudian tabel *positions* digunakan untuk menyimpan informasi posisi wawancara yang terdiri dari *role*, *level*, serta deskripsi posisi.



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

Pada proses wawancara, tabel `interview_sessions` digunakan untuk menyimpan data sesi wawancara yang sedang atau telah dilakukan oleh kandidat. Setiap sesi wawancara dapat memiliki beberapa pertanyaan yang tersimpan pada tabel `interview_questions`. Pertanyaan tersebut dihasilkan oleh sistem AI berdasarkan *role* dan tingkat kesulitan yang telah ditentukan.

Selanjutnya jawaban dari kandidat akan disimpan pada tabel `interview_answers`, yang berisi jawaban kandidat, skor penilaian, serta feedback yang diberikan oleh sistem. Hubungan antar tabel ini memungkinkan sistem untuk mengelola proses wawancara secara terstruktur mulai dari pemilihan posisi, proses wawancara, hingga evaluasi hasil wawancara.

4.3.5 Perancangan Antarmuka

Tahap ini merancang antarmuka sebelum proses implementasi antarmuka ke dalam basis website. Perancangan antarmuka bertujuan untuk memberikan gambaran visual mengenai tata letak, alur interaksi, dan komponen-komponen yang akan digunakan pengguna dalam menjalankan simulasi wawancara kerja berbasis AI. Perancangan ini mengacu pada alur proses bisnis sistem yang telah dijelaskan sebelumnya, meliputi autentikasi pengguna, pemilihan studi kasus, pelaksanaan simulasi wawancara, hingga penyajian hasil evaluasi jawaban kandidat.

1. Halaman Login

The image shows a login form with the following elements:

- A circular placeholder for a profile picture at the top.
- Greeting text: "Selamat Datang!"
- Instruction: "Silahkan masuk ke akun anda"
- Input field for "Nama Pengguna" (Username).
- Input field for "Kata Sandi" (Password).
- A "Masuk" (Login) button.
- A link: "Belum punya akun? Daftar di sini" (Don't have an account? Register here).

Gambar 4.15 Rancangan Halaman Login

Rancangan antarmuka pada Gambar 4.15 merupakan Halaman Login, yang berfungsi sebagai menu autentikasi hak akses pengguna sebelum memasuki sistem



© 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

utama. Pada halaman ini, pengguna diwajibkan untuk memasukkan Nama Pengguna dan Kata Sandi yang valid. Sistem juga menyediakan opsi navigasi registrasi di bagian bawah untuk mengarahkan pengguna ke halaman pembuatan akun jika mereka belum terdaftar.

2. Halaman Register

Gambar 4.16 Rancangan Halaman Register

Rancangan antarmuka pada Gambar 4.16 merupakan Halaman Register, yang digunakan untuk mengumpulkan data kredensial awal dari pengguna baru. Pada halaman ini, sistem mewajibkan pengisian identitas berupa Nama Pengguna dan Alamat Email, serta pembuatan Kata Sandi yang divalidasi ulang melalui kolom Konfirmasi Kata Sandi. Menu ini bertujuan untuk mendaftarkan hak akses baru yang nantinya akan disimpan secara aman di dalam basis data sistem.

3. Halaman Memilih Pekerjaan

Gambar 4.17 Rancangan Halaman Memilih Pekerjaan



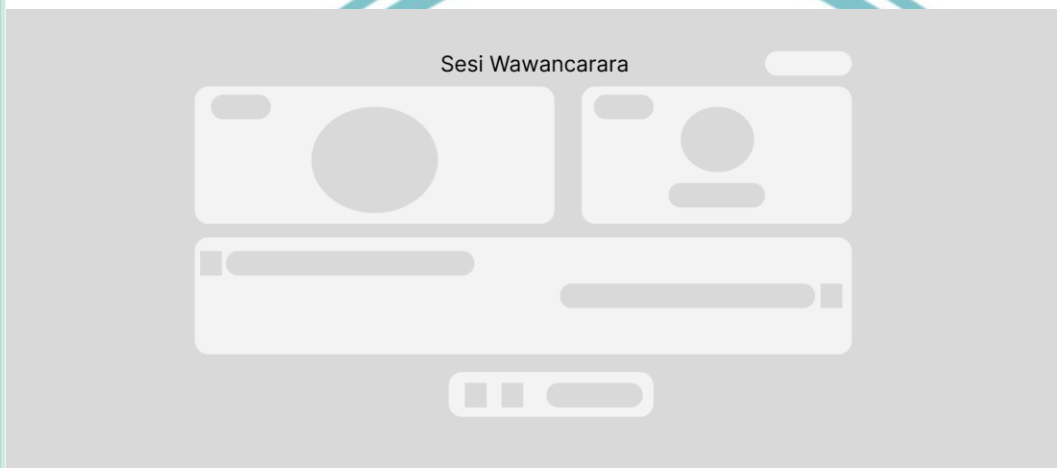
© Hak Cipta milik Politeknik Negeri Jakarta

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

Rancangan antarmuka pada Gambar 4.17 merupakan Halaman Pilih Posisi Interview yang berfungsi sebagai menu input kriteria simulasi. Halaman ini bertugas mengumpulkan data spesifik berupa Category, Role, dan Level pekerjaan dari pengguna. Seluruh data masukan ini menjadi parameter utama bagi sistem untuk memproses dan menyesuaikan topik pertanyaan sebelum pengguna menekan tombol "Mulai Wawancara".

4. Halaman Sesi Wawancara

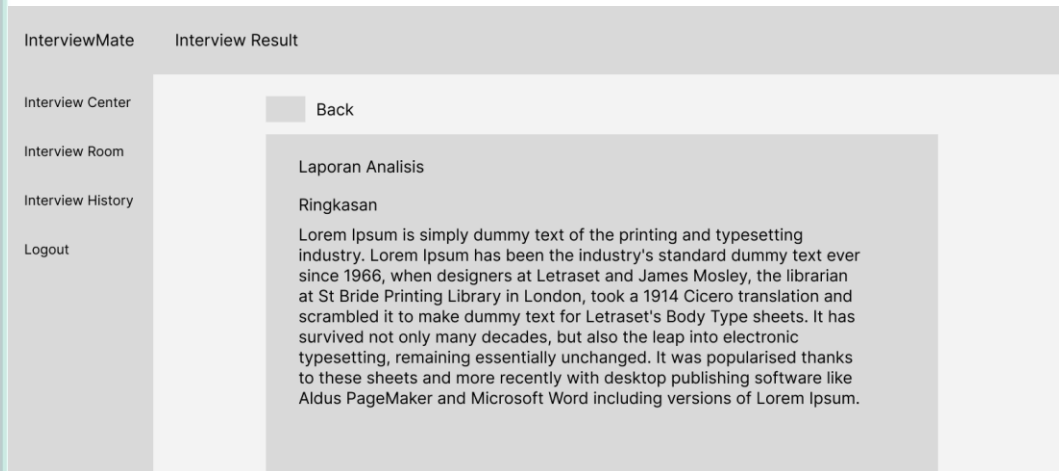


Gambar 4.18 Rancangan Halaman Sesi Wawancara

Rancangan antarmuka pada Gambar 4.18 merupakan Halaman Sesi Wawancara, yang berfungsi sebagai media interaksi utama dalam pelaksanaan simulasi wawancara kerja. Halaman ini dirancang untuk memproses dan menampilkan data komunikasi dua arah secara real-time, yang meliputi tampilan visual pewawancara berbasis AI dan respons video dari pengguna. Selain itu, halaman ini mengelola data transkrip percakapan dalam bentuk teks chat serta menyediakan kontrol audio-video di bagian bawah untuk mengatur jalannya sesi wawancara.



5. Halaman Hasil Evaluasi



Gambar 4.19 Rancangan Halaman Hasil Evaluasi

Rancangan antarmuka pada Gambar 4.19 merupakan Halaman Ringkasan dan Transkrip Wawancara, yang berfungsi untuk menyajikan dokumentasi tertulis dari sesi simulasi yang telah dilakukan. Halaman ini memuat dua bagian data utama, yaitu ringkasan yang berisi kesimpulan atau poin-poin penting performa pengguna, serta transcript yang menampilkan seluruh teks percakapan utuh antara pewawancara AI dan pengguna. Penyajian data ini bertujuan agar pengguna dapat meninjau kembali setiap detail jawaban mereka secara mendalam.

4.3 Implementasi Sistem

Pada tahap ini dilakukan implementasi sistem wawancara kerja berbasis *Retrieval-Augmented Generated (RAG)* yang telah dirancang pada tahap sebelumnya. Proses implementasi meliputi pengumpulan basis data pertanyaan wawancara, penerapan metode RAG, serta pengembangan antarmuka pengguna.

4.3.1 Implementasi Pengumpulan Basis Pengetahuan

1. Sumber Dataset

Pada tahap implementasi, dataset yang sebelumnya telah diidentifikasi pada tahap analisis kebutuhan digunakan sebagai sumber *knowledge base* sistem. Dataset yang digunakan berasal dari platform Kaggle, yaitu HR Interview Questions and *Ideal answers* untuk pertanyaan non-teknis dan behavioral, serta Software Engineering Interview Questions Dataset untuk pertanyaan teknis. Selain itu, digunakan pula dataset tambahan berisi kumpulan pertanyaan wawancara teknis dan behavioral untuk memperkaya variasi pertanyaan 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 mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin 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

2. Karakteristik Dataset

Dataset yang digunakan memiliki format yang berbeda, yaitu CSV dan JSON, dengan struktur atribut yang belum seragam. Beberapa atribut penting yang digunakan meliputi *question*, *answer/ideal_answer*, *role*, *difficulty*, *category/topic*, serta *experience*.

3. Pembersihan Data

Setelah data didapat, selanjutnya dilakukan proses pembersihan data untuk meningkatkan konsistensi dan kualitas dataset sebelum digunakan sebagai *knowledge base*. Tahapan ini meliputi penghapusan spasi berlebih, penyeragaman nama kolom, pembersihan nilai kategori, serta penyesuaian atribut yang memiliki penamaan berbeda antar dataset.

4. Transformasi Data

Setelah proses pembersihan, dilakukan transformasi data agar seluruh dataset memiliki struktur yang sama. Tahap ini mencakup pemetaan kategori pertanyaan ke dalam *role* target, seperti Frontend Developer, Backend Developer, Software Engineer, DevOps Engineer, dan Data Scientist. Selain itu, atribut tingkat kesulitan dan pengalaman kerja juga dikonversi ke dalam level senioritas, yaitu Junior, Intermediate, dan Senior.

```
# proses pengelompokan
def refine_role(row):
    role = row['role']
    text = (str(row['question']) + " " + str(row['ideal_answer'])).lower()

    # role software engineer
    if role == 'Software Engineer':
        if any(x in text for x in ['frontend', 'front-end', 'react', 'vue', 'angular', 'css', 'html', 'javascript', 'ui/ux']):
            return 'Frontend Developer'
        elif any(x in text for x in ['backend', 'back-end', 'api', 'sql', 'database', 'server', 'java', 'python', 'node', 'django']):
            return 'Backend Developer'
        else:
            return 'Software Engineer'
    # role data scientist
    elif role == 'Data Scientist':
        if any(x in text for x in ['machine learning', 'deep learning', 'neural network', 'ai', 'nlp', 'computer vision', 'model training']):
            return 'AI/ML Engineer'
        elif any(x in text for x in ['etl', 'pipeline', 'big data', 'hadoop', 'spark', 'kafka', 'warehouse', 'data engineering']):
            return 'Data Engineer'
        elif any(x in text for x in ['visualization', 'dashboard', 'tableau', 'power bi', 'excel', 'reporting', 'business insight', 'analytics']):
            return 'Data Analyst'
        else:
            return 'Data Scientist'

    # mengubah nama QA analis menjadi QA engineer
    elif role == 'QA Analyst':
        return 'QA Engineer'

    # sisanya menampilkan role yang tidak dipecah atau ubah
    return role
```

Gambar 4.20 Transformasi Data

Gambar 4.20 menunjukkan proses transformasi data yang dilakukan untuk menyeragamkan kategori pertanyaan ke dalam *role* yang lebih spesifik menggunakan aturan berbasis kata kunci (rule-based mapping). Pada proses ini, sistem membaca informasi *role*, *question*, dan *ideal_answer*, kemudian



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

mengelompokkannya ke dalam *role* yang sesuai, seperti Frontend Developer, Backend Developer, Data Engineer, Data Analyst, dan QA Engineer. Transformasi ini bertujuan agar data memiliki struktur yang konsisten sehingga dapat digunakan pada tahap pelatihan dan pengolahan berikutnya.

5. Penggabungan Dataset

Setelah masing – masing dataset dibersihkan dan ditransformasikan, seluruh data digabungkan menjadi satu *knowledge base*. Proses ini melalui pendekatan schema mapping, yaitu penyelarasan atribut antar dataset ke dalam struktur akhir yang seragam, seperti Question, Answer, Role, Topic, Difficulty, Seniority, dan Stage.

```
# standarisasi role
def standardize_role_name(role_name):
    role_name = str(role_name).strip().lower()

    mapping = {
        'software engineering': 'Software Engineer',
        'software engineer': 'Software Engineer',
        'backend development': 'Backend Developer',
        'backend developer': 'Backend Developer',
        'frontend development': 'Frontend Developer',
        'frontend developer': 'Frontend Developer',
        'web development': 'Frontend Developer',
        'devops engineering': 'DevOps Engineer',
        'devops engineer': 'DevOps Engineer',
        'qa engineering': 'QA Engineer',
        'qa engineer': 'QA Engineer',
        'qa analyst': 'QA Engineer',
        'data science': 'Data Scientist',
        'data scientist': 'Data Scientist',
        'ai/ml engineer': 'Data Scientist',
        'data engineer': 'Data Scientist'
    }

    return mapping.get(role_name, role_name.title())

for df in [df_hr, df_sw, df_full]:
    df['Role'] = df['Role'].apply(standardize_role_name)

# merge dataset
knowledge_df = pd.concat([df_hr, df_sw, df_full], ignore_index=True)

target_columns = [
    'Question', 'Answer', 'Role',
    'Topic', 'Difficulty', 'Seniority',
    'Stage', 'Source'
]

knowledge_df = knowledge_df[target_columns]
```

Gambar 4.21 Penggabungan Dataset

Gambar 4.21 menunjukkan proses penggabungan dataset yang diawali dengan standarisasi nama *role* untuk menyamakan variasi penamaan dari berbagai sumber data ke dalam kategori yang konsisten. Setelah proses penyelarasan selesai, seluruh dataset digabungkan (merge dataset) menjadi satu *knowledge base* menggunakan



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

pendekatan schema mapping. Pada tahap ini dipilih atribut akhir yang digunakan, yaitu Question, Answer, *Role*, Topic, *Difficulty*, Seniority, *Stage*, dan Source, sehingga seluruh data memiliki format yang seragam untuk digunakan pada tahap pemrosesan berikutnya.

6. Penghapusan Data Duplikat

Proses ini dilakukan untuk menghapus pertanyaan yang memiliki isi serupa atau identik pada *role* yang sama. Langkah ini bertujuan untuk menjaga kualitas *knowledge base* serta meningkatkan variasi pertanyaan yang dapat digunakan sistem selama proses wawancara.

```
# remove duplicates pertanyaan
knowledge_df['Question_Lower'] = knowledge_df['Question'].str.lower().str.strip()
knowledge_df.drop_duplicates(subset=['Question_Lower', 'Role'], inplace=True)
knowledge_df.drop(columns=['Question_Lower'], inplace=True)
```

Gambar 4.22 Penghapusan Data Duplikat

Gambar 4.22 menunjukkan proses penghapusan data duplikat yang dilakukan dengan melakukan normalisasi teks pertanyaan terlebih dahulu, yaitu mengubah seluruh huruf menjadi huruf kecil (lowercase) dan menghapus spasi yang tidak diperlukan. Setelah itu, sistem melakukan pemeriksaan duplikasi berdasarkan kombinasi atribut Question dan *Role*, sehingga pertanyaan yang memiliki isi identik pada *role* yang sama hanya disimpan satu kali. Tahap ini bertujuan untuk menjaga kualitas *knowledge base* dan mengurangi redundansi data.

7. Penambahan Metadata

Proses ini dilakukan untuk mendukung mekanisme *retrieval* yang lebih kontekstual melalui penambahan metadata atau smart tagging pada setiap pertanyaan. Metadata yang ditambahkan meliputi tahapan wawancara (*Stage*), jenis pertanyaan (*Question Type*), area kompetensi (*Competency Area*), level adaptif (*Adaptive Level*), dan penanda pertanyaan akhir. Penambahan metadata ini bertujuan agar sistem dapat memilih pertanyaan yang lebih sesuai dengan posisi, tingkat kemampuan, serta alur wawancara 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 mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
def classify_question_type(text):
    if any(x in text for x in ['design', 'architecture', 'scalability', 'trade-off', 'system']):
        return 'System Design'
    if any(x in text for x in ['algorithm', 'complexity', 'optimize', 'solve']):
        return 'Problem Solving'
    if any(x in text for x in ['conflict', 'challenge', 'failure', 'proud']):
        return 'Behavioral'
    if any(x in text for x in ['lead', 'manage', 'mentor']):
        return 'Leadership'
    return 'Technical Knowledge'

knowledge_df['Question_Type'] = knowledge_df['Combined_Text'].apply(classify_question_type)

# competency area
def classify_competency(text):
    if any(x in text for x in ['sql', 'database', 'mongo', 'postgres', 'index']):
        return 'Database'
    if any(x in text for x in ['api', 'rest', 'http', 'json', 'protocol']):
        return 'Networking/API'
    if any(x in text for x in ['react', 'css', 'html', 'dom', 'frontend']):
        return 'Frontend'
    if any(x in text for x in ['docker', 'kubernetes', 'ci/cd', 'aws', 'cloud']):
        return 'DevOps'
    if any(x in text for x in ['test', 'junit', 'mock', 'qa']):
        return 'Testing'
    return 'General Engineering'

knowledge_df['Competency_Area'] = knowledge_df['Combined_Text'].apply(classify_competency)

# adaptive level
def map_adaptive_level(val):
    val = str(val).lower()
    if 'easy' in val:
        return 1
    if 'hard' in val or 'expert' in val:
        return 3
    return 2

knowledge_df['Adaptive_Level'] = knowledge_df['Combined_Text'].apply(map_adaptive_level)
```

Gambar 4.23 Penambahan Metadata

Gambar 4.23 menunjukkan proses penambahan metadata (*smart tagging*) pada dataset dengan memanfaatkan aturan berbasis kata kunci (*rule-based classification*). Pada tahap ini, sistem mengklasifikasikan setiap pertanyaan ke dalam beberapa atribut tambahan, yaitu *Question Type*, *Competency Area*, dan *Adaptive Level*. *Question Type* digunakan untuk mengelompokkan jenis pertanyaan, seperti *System Design*, *Problem Solving*, *Behavioral*, *Leadership*, atau *Technical Knowledge*. Selanjutnya *Competency Area* digunakan untuk mengidentifikasi bidang kompetensi yang diuji, seperti *Database*, *Networking/API*, *Frontend*, *DevOps*, dan *Testing*. Kemudian *Adaptive Level* ditambahkan untuk merepresentasikan tingkat kesulitan pertanyaan yang akan digunakan pada mekanisme wawancara adaptif.

8. Pembentukan Teks Embedding

Setelah data terstruktur, setiap entri pertanyaan dibentuk menjadi representasi teks khusus yang akan digunakan untuk proses *embedding*. Representasi ini menggabungkan informasi penting, seperti *role*, *stage*, tingkat kesulitan, jenis



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

pertanyaan, area kompetensi, pertanyaan, dan jawaban ideal. Penggabungan atribut ini dilakukan agar proses pencarian semantik dapat mempertimbangkan konteks pertanyaan secara lebih lengkap.

```
# embed text
knowledge_df['Embedding_Text'] = (
    "Role: " + knowledge_df['Role'].fillna('') + ". " +
    "Stage: " + knowledge_df['Stage'].fillna('') + ". " +
    "Level: " + knowledge_df['Adaptive_Level'].astype(str) + ". " +
    "Type: " + knowledge_df['Question_Type'].fillna('') + ". " +
    "Competency: " + knowledge_df['Competency_Area'].fillna('') + ". " +
    "Question: " + knowledge_df['Question'].fillna('') + ". " +
    "Ideal Answer: " + knowledge_df['Answer'].fillna('')
)

knowledge_df.drop(columns=['Combined_Text'], inplace=True)
```

Gambar 4.24 Pembentukan Teks Embedding

Gambar 4.24 menunjukkan proses pembentukan teks embedding dengan menggabungkan beberapa atribut yang telah diproses sebelumnya ke dalam satu representasi teks terstruktur. Atribut yang digabungkan meliputi *role*, *stage*, *difficulty*, *question type*, *competency area*, *question*, dan *ideal answer*. Penggabungan informasi tersebut dilakukan agar proses embedding dapat merepresentasikan tidak hanya isi pertanyaan, tetapi juga konteks dan karakteristik pertanyaan secara lebih menyeluruh. Hasil representasi ini kemudian digunakan pada proses *semantic retrieval* untuk menghasilkan pencarian pertanyaan yang lebih relevan.

9. Penyimpanan ke *vector database*

Data akhir yang telah diproses kemudian diubah ke dalam bentuk embedding menggunakan model *Sentence Transformer* dengan arsitektur *paraphrase-multilingual-MiniLM-L12-v2*. Hasil embedding disimpan ke dalam *vector database* menggunakan ChromaDB. Penyimpanan ini memungkinkan sistem melakukan pencarian berbasis kemiripan semantik, sehingga pertanyaan yang diambil tidak hanya bergantung pada pencocokan kata kunci, tetapi juga makna konteksnya. sebagaimana diilustrasikan pada Gambar 4.25



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

```
global chroma_client, collection

if collection:
    return collection

chroma_path = os.path.join(
    PROJECT_ROOT,
    'chroma_storage'
)

chroma_client = chromadb.PersistentClient(
    path=chroma_path
)

embed_fn = embedding_functions.SentenceTransformerEmbeddingFunction(
    model_name="paraphrase-multilingual-MiniLM-L12-v2"
)

collection = chroma_client.get_or_create_collection(
    name="adaptive_interview_rag",
    embedding_function=embed_fn
)

if collection.count() == 0:

    if not os.path.exists(DATA_PATH):
        return None

    print("Memproses CSV ke ChromaDB...")
```

Gambar 4.25 Penyimpanan Data ke Vector Database

4.3.2 Implementasi Retrieval-Augmented Generation (RAG)

Implementasi metode *Retrieval-Augmented Generation* (RAG) pada sistem ini dibangun melalui tiga tahapan utama, yaitu *retrieval*, *augmentation*, dan *generation*. Ketiga tahapan tersebut saling terhubung untuk memungkinkan sistem mengambil informasi yang relevan dari *knowledge base*, menyusunnya sebagai konteks, dan menghasilkan pertanyaan maupun evaluasi wawancara secara adaptif.

1. Retrieval (Pengambilan Data)

Tahap *retrieval* menunjukkan proses pengambilan data yang relevan dari *knowledge base* untuk mendukung jalannya sesi wawancara secara adaptif. Tahap ini berperan penting dalam mekanisme *Retrieval-Augmented Generation* (RAG), karena sistem harus mampu menemukan pertanyaan yang sesuai dengan konteks pengguna sebelum pertanyaan tersebut diproses lebih lanjut oleh model bahasa.



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

Dalam implementasinya, proses *retrieval* pada sistem ini pengambilan pertanyaan secara adaptif, yaitu proses pencarian data pertanyaan yang paling relevan berdasarkan konteks sesi wawancara yang sedang berlangsung. Pencarian dilakukan dengan mempertimbangkan beberapa parameter, seperti posisi yang dipilih pengguna, tahapan wawancara, tingkat kesulitan, serta riwayat pertanyaan yang telah digunakan sebelumnya. Dengan mekanisme ini, sistem dapat memilih pertanyaan yang lebih sesuai, terarah, dan tidak berulang selama proses wawancara berlangsung. Implementasi proses pengambilan data ditunjukkan pada Gambar 4.26.

```
def retrieve_adaptive_question(role, stage, difficulty, used_ids):
    global collection
    if not collection:
        init_knowledge_base()

    where_filter = {"$and": [{"role": role}, {"stage": stage}]}

    if stage == "Technical":
        where_filter["$and"].append({"difficulty_level": difficulty})

    results = collection.query(
        query_texts=[f"{role} {stage} level {difficulty}"],
        n_results=20,
        where=where_filter
    )

    if not results["documents"] or not results["documents"][0]:
        return None, None, None

    for i, doc in enumerate(results["documents"][0]):
        rag_id = results["ids"][0][i]
        if rag_id not in used_ids:
            return doc, results["metadatas"][0][i], rag_id

    return None, None, None
```

Gambar 4.26 Fungsi Adaptive Question

Gambar 4.26 menunjukkan fungsi `retrieve_adaptive_question` yang bertugas mengambil pertanyaan wawancara berdasarkan parameter `role`, `stage`, tingkat kesulitan, dan daftar ID pertanyaan yang sudah digunakan (`used_ids`). Pertama, fungsi akan memeriksa apakah instance koleksi data (`collection`) sudah termuat di memori. Apabila koneksi belum terbentuk, fungsi akan memanggil `init_knowledge_base()` untuk memuat dan membuka koneksi ke vector database yang telah menyimpan data embedding sebelumnya.

Selanjutnya, sistem membuat filter untuk menyaring data berdasarkan `role` dan `stage`. Jika tahapnya adalah `Technical`, maka filter akan ditambahkan lagi 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 mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

kondisi tingkat kesulitan agar hasilnya lebih spesifik. Setelah itu, fungsi melakukan pencarian ke dalam database menggunakan *collection.query*, dengan *query* berupa teks yang menggabungkan *role*, *stage*, dan *difficulty*. Fungsi ini akan mengambil hingga 20 hasil yang paling relevan.

Jika tidak ada hasil yang ditemukan, fungsi langsung mengembalikan nilai None. Namun jika ada hasil, fungsi akan melakukan perulangan pada setiap dokumen yang ditemukan. Di setiap iterasi, fungsi akan mengecek apakah ID pertanyaan tersebut sudah pernah digunakan atau belum. Jika belum pernah digunakan, maka pertanyaan tersebut akan langsung dikembalikan bersama metadata dan ID-nya.

2. Augmentation (Penggabungan Konteks)

Tahap *Augmentation* menunjukkan proses penyusunan instruksi yang memanfaatkan seluruh informasi hasil *retrieval*. Tahap ini berperan untuk memberikan informasi yang jelas dan landasan konteks yang komprehensif bagi model bahasa. Implementasi dari proses penggabungan data ini dapat dilihat pada Gambar 4.27.

```
doc, meta, rag_id = retrieve_adaptive_question(
    f"{session.level} {session.role_focus}",
    session.current_stage,
    session.current_difficulty,
    used_ids
)

ideal = meta["answer"] if meta else "Jawaban sesuai best practice."
context = doc if doc else "Konsep umum."

diff_label = ["Easy", "Medium", "Hard"][session.current_difficulty - 1]

past_questions_records = InterviewQuestion.query.filter_by(session_id=session_id).order_by(
    InterviewQuestion.order_num.asc()).all()
recent_questions = [q.question_text for q in past_questions_records[-4:]] if
past_questions_records else []

next_question = generate_interview_question(
    context_text=context,
    ideal_answer=ideal,
    role=f"{session.level} {session.role_focus}",
    difficulty_label=diff_label,
    stage=session.current_stage,
    previous_questions=recent_questions
)
```

Gambar 4.27 Penggabungan Konteks

Gambar 4.27 menunjukkan implementasi tahap augmentation pada sistem RAG, yaitu proses penggabungan hasil *retrieval* dengan konteks tambahan sebelum dikirim ke model bahasa. Setelah sistem memperoleh dokumen yang relevan dari *knowledge base* melalui fungsi *retrieve_adaptive_question()*, informasi tersebut



© 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

tidak langsung ditampilkan, melainkan diperkaya terlebih dahulu dengan beberapa elemen pendukung. Konteks yang digabungkan meliputi isi dokumen hasil *retrieval*, jawaban referensi (*ideal answer*) yang tersimpan pada metadata, level kandidat, posisi yang dipilih, tahapan wawancara, tingkat kesulitan pertanyaan, serta riwayat pertanyaan sebelumnya.

Penggabungan konteks ini bertujuan agar model bahasa tidak hanya menghasilkan pertanyaan berdasarkan satu topik utama, tetapi juga mempertimbangkan alur wawancara yang sedang berlangsung. Dengan demikian, pertanyaan yang dihasilkan menjadi lebih relevan, tidak berulang, serta lebih sesuai dengan kemampuan dan progres kandidat selama sesi wawancara. Tahap *augmentation* ini menjadi bagian penting dalam memastikan bahwa sistem mampu membangun interaksi wawancara yang lebih adaptif dan kontekstual.

3. Generation (Pembangkitan Jawaban)

Tahap *generation* merupakan proses yang memanfaatkan model bahasa untuk menghasilkan keluaran akhir berdasarkan konteks yang telah dibentuk pada tahap sebelumnya. Dalam penelitian ini, proses pembangkitan tidak hanya digunakan untuk membuat pertanyaan wawancara, tetapi juga untuk melakukan evaluasi terhadap jawaban pengguna serta menyusun laporan akhir hasil wawancara.

Terdapat tiga bentuk pembangkitan utama pada sistem ini. Pertama, model digunakan untuk menghasilkan pertanyaan wawancara yang lebih natural dan relevan berdasarkan hasil *retrieval*, metadata, serta konteks sesi wawancara yang sedang berlangsung. Kedua, proses melakukan evaluasi jawaban kandidat, yaitu dengan membandingkan jawaban pengguna terhadap jawaban referensi yang tersedia dalam *knowledge base*.

Ketiga, model juga digunakan untuk menyusun laporan akhir wawancara secara otomatis setelah sesi selesai. Laporan ini berisi rangkuman performa kandidat, kekuatan yang ditunjukkan selama wawancara, kelemahan yang masih perlu ditingkatkan, serta rekomendasi pembelajaran yang relevan.



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

a. Generate Pertanyaan

```
previous_questions_instruction = ""
if previous_questions:
    previous_questions_instruction = f"\nPENTING: JANGAN ulangi atau gunakan topik yang mirip dengan
    daftar pertanyaan sebelumnya berikut ini:\n- " + "\n- ".join(previous_questions)

if stage == "Study Case":
    tugas = "Berikan sebuah studi kasus (Study Case) atau skenario problem-solving berskala
    arsitektur/sistem."
elif stage == "Soft Skill":
    tugas = "Berikan pertanyaan wawancara tentang Soft Skill (komunikasi, penyelesaian konflik,
    manajemen waktu)."
else:
    tugas = "Buat 1 pertanyaan wawancara teknis spesifik."

system_instruction = f"""Anda adalah Senior Tech Interviewer berbahasa Indonesia untuk posisi {role}
(Level: {difficulty_label}).

Topik Inti (Bahan Pertanyaan): {context_text}
Ekspektasi Jawaban: {ideal_answer}
{previous_questions_instruction}

Instruksi:
1. Tugas Anda: {tugas}
{bg_instruction}
2. Fokus pada 'Topik Inti'.
3. Langsung berikan pertanyaan tanpa kata pengantar, sapaan, atau basa-basi.
"""

try:
    res = client.chat.completions.create(
        model="llama-3.3-70b-versatile",
        messages=[{"role": "user", "content": system_instruction}],
```

Gambar 4.28 Generate Pertanyaan

Gambar 4.28 merupakan proses generation pembuatan pertanyaan wawancara menggunakan model bahasa Llama 3.3 70B Versatile. Pada tahap ini, sistem menyusun sebuah instruksi terstruktur yang berfungsi sebagai panduan bagi model bahasa agar dapat menghasilkan pertanyaan wawancara yang adaptif. Konteks masukan tersebut memuat beberapa informasi penting, seperti posisi yang dilamar, tingkat kesulitan pertanyaan, topik inti hasil retrieval, ekspektasi jawaban, serta riwayat pertanyaan sebelumnya agar sistem tidak menghasilkan pertanyaan yang berulang.

Selain itu, sistem juga memberikan instruksi tambahan kepada model agar pertanyaan yang dihasilkan tetap fokus pada topik yang telah ditentukan dan disampaikan secara langsung tanpa sapaan atau kalimat pembuka yang tidak diperlukan. Setelah kerangka perintah tersebut selesai disusun, sistem mengirimkannya secara langsung ke model bahasa Llama-3.3-70B-Versatile. Model kemudian memproses seluruh parameter tersebut untuk merumuskan sebuah



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

pertanyaan baru dalam bentuk teks, yang nantinya akan langsung digunakan sebagai pertanyaan selanjutnya di sesi wawancara.

b. Evaluasi Jawaban Kandidat

Tahap evaluasi jawaban pada sistem ini dimulai dengan melakukan analisis kuantitatif terhadap jawaban yang diberikan oleh kandidat. Proses ini bertujuan untuk mengukur tingkat kesesuaian antara jawaban kandidat dengan jawaban referensi. Implementasi dari algoritma perhitungan ini ditunjukkan pada Gambar 4.29.

```
def compute_tfidf_vector(tf: dict, idf: dict, vocab: list[str]) -> list[float]:
    return [tf.get(term, 0.0) * idf.get(term, 0.0) for term in vocab]

def calculate_cosine_similarity(vector_a, vector_b) -> float:
    dot_product = sum(a * b for a, b in zip(vector_a, vector_b))
    magnitude_a = math.sqrt(sum(a ** 2 for a in vector_a))
    magnitude_b = math.sqrt(sum(b ** 2 for b in vector_b))
    if magnitude_a == 0 or magnitude_b == 0:
        return 0.0
    return dot_product / (magnitude_a * magnitude_b)
```

Gambar 4.29 Proses Perhitungan Kecocokan Jawaban

Gambar 4.29 merupakan implementasi dari dua fungsi inti dalam proses perhitungan kecocokan antara jawaban kandidat dengan jawaban referensi secara matematis, yaitu fungsi `compute_tfidf_vector` dan `calculate_cosine_similarity`.

Fungsi `compute_tfidf_vector` bertugas membentuk vektor bobot TF-IDF dari sebuah dokumen (baik jawaban referensi maupun jawaban kandidat). Fungsi ini menerima nilai *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) yang telah dihitung sebelumnya, kemudian mengalikan keduanya ($TF \times IDF$) untuk setiap term pada daftar *vocabulary* gabungan, sehingga menghasilkan sebuah vektor numerik yang merepresentasikan bobot penting setiap kata dalam dokumen tersebut.

Selanjutnya, fungsi `calculate_cosine_similarity` digunakan untuk mengukur tingkat kemiripan antara dua vektor TF-IDF (vektor jawaban referensi dan vektor jawaban kandidat) menggunakan rumus *Cosine Similarity*. Secara matematis, nilai *cosine similarity* dihitung dengan membagi hasil perkalian titik (*dot product*) antara dua vektor dengan perkalian panjang (*magnitude*) masing-masing vektor. Fungsi ini juga menangani kondisi khusus ketika salah satu vektor memiliki *magnitude*



© 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

bernilai 0 yaitu ketika sebuah dokumen tidak memiliki term yang overlap sama sekali dengan *vocabulary*, yang secara otomatis akan mengembalikan nilai kemiripan 0 untuk menghindari pembagian dengan nol.

Nilai yang dihasilkan berkisar antara 0 hingga 1, di mana semakin mendekati 1 berarti jawaban kandidat semakin mirip secara leksikal dan bobot kata kuncinya dengan jawaban referensi. Berbeda dengan pendekatan berbasis sentence embedding yang mengukur kemiripan makna kalimat secara semantik, pendekatan TF-IDF ini mengukur kemiripan berdasarkan kesamaan kata kunci yang muncul pada kedua dokumen beserta bobot kepentingannya masing-masing, sehingga lebih menitikberatkan pada seberapa banyak istilah teknis yang sama-sama disebutkan oleh kandidat dan referensi.

```
return {
  "score": round(final_score * 100),
  "feedback": feedback,
  "status": status,
  "cosine_detail": {
    "keyword_coverage_pct": float(round(cosine_data["keyword_coverage"] * 100, 1)),
    "keywords_found": cosine_data["keywords_found"],
    "keywords_missing": cosine_data["missing_keywords"],
    "tfidf_cosine_score": float(cosine_data["tfidf_cosine_score"]),
  }
}
```

Gambar 4.30 Pengembalian Objek Hasil Evaluasi

Gambar 4.30 merupakan proses pengumpulan seluruh hasil evaluasi ke dalam satu objek yang dikembalikan oleh sistem. Objek ini terdiri dari skor akhir, feedback, status kualitas jawaban, serta *cosine_detail* yang mencakup persentase keyword yang berhasil ditemukan dalam jawaban kandidat, daftar keyword yang muncul maupun yang tidak muncul, dan skor TF-IDF *cosine similarity* secara keseluruhan. Data inilah yang kemudian diterima oleh antarmuka untuk ditampilkan sebagai umpan balik kepada kandidat.



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

c. Generate Laporan Akhir

```
blocks = []

for i, h in enumerate(history, 1):
    block = (
        f"[Pertanyaan {i}] Stage: {h.get('stage', '-')}\n"
        f"  Q : {h['question']}\n"
        f"  A : {h['user_answer']}"
    )
    blocks.append(block)

data_str = "\n\n".join(blocks)

system_instruction = f"""Buatlah ringkasan evaluasi wawancara kerja untuk kandidat posisi {role}
berdasarkan data wawancara berikut:
```

Gambar 4.31 Generate Laporan Akhir

Gambar 4.31 menunjukkan proses penyusunan konteks yang digunakan untuk menghasilkan laporan akhir wawancara. Fungsi menerima parameter history tanya jawab wawancara dan posisi pekerjaan yang dilamar. Selanjutnya sistem melakukan iterasi terhadap setiap riwayat wawancara menggunakan enumerate, kemudian menyusun tahap wawancara, pertanyaan, dan jawaban kandidat ke dalam blok teks terstruktur. Seluruh blok tersebut digabungkan menjadi satu string yang merepresentasikan keseluruhan riwayat wawancara. String tersebut kemudian disisipkan ke dalam system instruction sebagai konteks bagi model bahasa untuk menghasilkan ringkasan substansi wawancara, penilaian kelayakan teknis, dan analisis struktur berpikir kandidat.

4.3.3 Implementasi Penilaian Jawaban Kandidat

Tahap ini Tahap ini mengimplementasikan penilaian jawaban kandidat secara manual menggunakan jawaban referensi yang tersimpan pada ChromaDB untuk memvalidasi akurasi perhitungan TF-IDF dan *Cosine Similarity*. Data yang digunakan terdiri atas jawaban referensi dari ChromaDB dan jawaban kandidat yang diperoleh dari riwayat sesi wawancara. Selanjutnya, sistem menghitung tingkat kemiripan antara kedua jawaban melalui tahapan preprocessing, pembobotan TF-IDF, dan perhitungan *Cosine Similarity* untuk memastikan hasil penilaian yang konsisten dan akurat.



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. Data Pertanyaan dan Jawaban

Tabel 9 menampilkan data berupa pasangan pertanyaan, jawaban referensi yang diperoleh dari ChromaDB, serta jawaban kandidat yang diambil dari riwayat wawancara sebagai data pengujian.

Tabel 9 Data Pertanyaan dan Jawaban

Pertanyaan	Jawaban Referensi	Jawaban Kandidat
Apa yang membuat antarmuka (interface) menjadi komponen penting dalam pemrograman, dan bagaimana hal ini memengaruhi kemudahan perawatan serta pengembangan sistem yang kompleks?	Antarmuka penting dalam pemrograman karena memungkinkan abstraksi dan modularitas, memisahkan antarmuka dari implementasi, sehingga meningkatkan kemudahan dan kemudahan perawatan kode. Antarmuka juga membantu mengurangi ketergantungan antar komponen dan meningkatkan penggunaan kembali kode. Dengan demikian, antarmuka mempermudah pengembangan dan pemeliharaan sistem yang kompleks.	Baik, untuk antarmuka merupakan sebuah proses yang sangat penting dalam pemrograman. Antarmuka juga membantu mengurangi ketergantungan antar komponen dan meningkatkan penggunaan kembali kode.

2. Proses Term Frequency-Inverse Document Frequency (TF-IDF)

Metode *Term Frequency-Inverse Document Frequency* (TF-IDF) merupakan algoritma statistik yang digunakan untuk mengevaluasi seberapa penting suatu kata



© 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

tunggal (*term*) di dalam sebuah dokumen atau kumpulan dokumen (S et al., 2023). Sebelum perhitungan matematis dilakukan, data teks transkrip jawaban kandidat dan dokumen referensi wajib melalui tahapan text preprocessing agar menjadi data bersih yang siap diolah. Alur pemrosesan ini berjalan secara urut melalui tahapan berikut:

a. Case Folding

Tahap case folding merupakan proses awal text *preprocessing* yang bertujuan untuk menyeragamkan bentuk huruf dengan mengubah seluruh karakter alfabet menjadi huruf kecil (*lowercase*). Proses ini dilakukan agar tidak terdapat perbedaan antara huruf kapital dan huruf kecil yang dapat memengaruhi proses pembentukan term pada tahap selanjutnya. Hasil proses *case folding* terhadap jawaban referensi dan jawaban kandidat ditunjukkan pada Tabel 10.

Tabel 10 Case Folding

Jawaban Referensi	Jawaban Kandidat
antarmuka penting dalam pemrograman karena memungkinkan abstraksi dan modularitas, memisahkan antarmuka dari implementasi, sehingga meningkatkan kemudahan dan kemudahan perawatan kode. antarmuka juga membantu mengurangi ketergantungan antar komponen dan meningkatkan penggunaan kembali kode. dengan demikian, antarmuka mempermudah pengembangan dan pemeliharaan sistem yang kompleks.	baik, untuk antarmuka merupakan sebuah proses yang sangat penting dalam pemrograman. antarmuka juga membantu mengurangi ketergantungan antar komponendan meningkatkan penggunaan kembali kode.

Berdasarkan Tabel 10, seluruh huruf kapital pada jawaban referensi maupun jawaban kandidat telah berhasil diubah menjadi huruf kecil secara seragam tanpa mengubah isi atau makna kalimat. Hasil *case folding* ini kemudian digunakan sebagai masukan pada tahap *tokenizing*.

b. Tokenizing

Tahap tokenizing bertujuan untuk memecah kalimat menjadi kumpulan kata (*token*) sehingga setiap kata dapat diproses secara individual pada tahap berikutnya. Hasil proses tokenizing ditunjukkan pada Tabel 11.



© 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

Tabel 11 Tokenizing

Jawaban Referensi	Jawaban Kandidat
antarmuka-penting-dalam-pemrograman-karena-memungkinkan-abstraksi-dan-modularitas-memisahkan-antarmuka-dari-implementasi-sehingga-meningkatkan-kemudahan-dan-kemudahan-perawatan-kode-antarmuka-juga-membantu-mengurangi-ketergantungan-antar-komponen-dan-meningkatkan-penggunaan-kembali-kode-dengan-demikian-antarmuka-mempermudah-pengembangan-dan-pemeliharaan-sistem-yang-kompleks	baik-untuk-antarmuka-merupakan-sebuah-proses-yang-sangat-penting-dalam-pemrograman-antarmuka-juga-membantu-mengurangi-ketergantungan-antar-komponen-dan-meningkatkan-penggunaan-kembali-kode

Berdasarkan Tabel 11, jawaban referensi menghasilkan sebanyak n token, sedangkan jawaban kandidat menghasilkan sebanyak n token. Setiap token selanjutnya akan melalui proses filtering untuk menghilangkan kata-kata yang tidak memiliki kontribusi terhadap pembobotan TF-IDF.

c. Filtering

Tahap filtering dilakukan untuk menghilangkan kata-kata umum (stopwords) yang tidak memberikan kontribusi signifikan terhadap proses pembobotan TF-IDF. Hasil proses *filtering* ditunjukkan pada Tabel 12.

Tabel 12 Filtering

Jawaban Referensi	Jawaban Kandidat
antarmuka-penting-dalam-pemrograman-karena-memungkinkan-abstraksi-dan-modularitas-memisahkan-antarmuka-dari-implementasi-sehingga-meningkatkan-kemudahan-dan-kemudahan-perawatan-kode-antarmuka-juga-membantu-mengurangi-ketergantungan-antar-komponen-dan-meningkatkan-penggunaan-kembali-kode-dengan-demikian-antarmuka-mempermudah-pengembangan-dan-pemeliharaan-sistem-yang-kompleks	baik-untuk-antarmuka-merupakan-sebuah-proses-yang-sangat-penting-dalam-pemrograman-antarmuka-juga-membantu-mengurangi-ketergantungan-antar-komponen-dan-meningkatkan-penggunaan-kembali-kode



© 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

Berdasarkan Tabel 12, kata-kata seperti "dan", "yang", "dalam", serta stopword lainnya telah dihapus sehingga hanya tersisa kata-kata yang memiliki makna penting.

d. Stemming

Tahap stemming bertujuan mengubah setiap kata berimbuhan menjadi bentuk kata dasarnya sehingga variasi kata dapat direpresentasikan sebagai satu term yang sama. Hasil proses *stemming* ditunjukkan pada Tabel 13.

Tabel 13 Stemming

Jawaban Referensi	Jawaban Kandidat
antarmuka-penting-pemrograman-mungkin-abstraksi-modularitas-pisah-antarmuka-implementasi-tingkat-mudah-mudah-awat-kode-antarmuka-bantu-kurang-gantung-antar-komponen-tingkat-guna-kembali-kode-antarmuka-mudah-kembang-pelihara-sistem-kompleks	antarmuka-proses-penting-pemrograman-antarmuka-bantu-kurang-gantung-antar-komponen-tingkat-guna-kembali-kode

Berdasarkan Tabel 13, kata-kata berimbuhan telah berhasil dikonversi menjadi bentuk dasar, sehingga daftar term menjadi lebih konsisten untuk proses pembentukan *Term Frequency* (TF). Selanjutnya, seluruh term hasil *preprocessing* dari jawaban referensi dan jawaban kandidat digabungkan untuk membentuk daftar term yang digunakan sebagai dasar perhitungan *Term Frequency* (TF), *Document Frequency* (DF), *Inverse Document Frequency* (IDF), dan bobot TF-IDF. Daftar term hasil *preprocessing* ditunjukkan pada Tabel 14.

Tabel 14 Daftar Term Hasil Preprocessing

ti	Term
ti1	abstraksi
ti2	antar
ti3	antarmuka
ti4	awat
ti5	bantu
ti6	gantung
ti7	guna
ti8	implementasi



© 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

ti9	kembali
ti10	kembang
ti11	kode
ti12	kompleks
ti13	komponen
ti14	kurang
ti15	modularitas
ti16	mudah
ti17	mungkin
ti18	pelihara
ti19	pemrograman
ti20	penting
ti21	pisah
ti22	proses
ti23	sistem
ti24	tingkat

Berdasarkan Tabel 14, diperoleh sebanyak 24 term unik yang merupakan hasil penggabungan term dari jawaban referensi dan jawaban kandidat setelah melalui tahapan *text preprocessing*. Daftar term tersebut menjadi acuan dalam proses perhitungan *Term Frequency* (TF) pada masing-masing dokumen sehingga setiap term memiliki bobot yang dapat digunakan pada tahap perhitungan *Document Frequency* (DF), *Inverse Document Frequency* (IDF), hingga pembentukan vektor TF-IDF.

e. Term Frequency (TF)

Tahap *Term Frequency* (TF) bertujuan untuk menghitung tingkat kemunculan setiap term pada masing-masing dokumen hasil *preprocessing*. Pada penelitian ini digunakan *normalized term frequency*, yaitu frekuensi kemunculan suatu term dibandingkan dengan jumlah seluruh term dalam dokumen. Pendekatan ini menghasilkan bobot yang bersifat proporsional sehingga dapat mengurangi pengaruh perbedaan panjang dokumen terhadap proses pembobotan kata. Perhitungan nilai TF dilakukan menggunakan persamaan

$$TF(t, d) = \frac{f_{t,d}}{\sum_k f_{k,d}}$$

Keterangan:



© 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

- $TF(t, d)$ = nilai Term Frequency dari term t pada dokumen d
- ft, d = jumlah kemunculan term t pada dokumen d
- $\sum k f_{k,d}$ = jumlah keseluruhan term (total token hasil preprocessing) pada dokumen d

Oleh karena itu, nilai TF menunjukkan rasio kemunculan suatu term terhadap seluruh term dalam dokumen. Semakin sering suatu term muncul pada dokumen, maka semakin besar nilai TF yang dihasilkan. Berikut ini contoh perhitungan TF: Berdasarkan hasil *stemming*, jumlah *term* pada jawaban referensi (D1) sebanyak 30 term, sedangkan jumlah term pada jawaban kandidat (D2) sebanyak 14 term. Sebagai contoh, term *antarmuka* muncul sebanyak 4 kali pada dokumen D1 sehingga nilai TF dihitung sebagai berikut.

$$TF(\text{antarmuka}, D1) = \frac{4}{30} = 0,1333$$

Sementara itu, term *abstraksi* hanya muncul sebanyak 1 kali pada dokumen 1 sehingga diperoleh nilai TF sebagai berikut.

$$TF(\text{abstraksi}, D1) = \frac{1}{30} = 0,0333$$

Jika suatu term tidak muncul pada dokumen tertentu, maka nilai TF untuk term tersebut adalah 0. Hasil perhitungan TF seluruh term pada jawaban referensi dan jawaban kandidat ditunjukkan pada Tabel 15.

Tabel 15 Hasil Perhitungan TF

ti	Term	TF D1 (Referensi)	TF D2 (Kandidat)
ti1	abstraksi	0,0333	0
ti2	antar	0,0333	0,0714
ti3	antarmuka	0,1333	0,1429
ti4	awat	0,0333	0
ti5	bantu	0,0333	0,0714
ti6	gantung	0,0333	0,0714
ti7	guna	0,0333	0,0714
ti8	implementasi	0,0333	0
ti9	kembali	0,0333	0,0714
ti10	kembang	0,0333	0
ti11	kode	0,0667	0,0714
ti12	kompleks	0,0333	0



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

ti13	komponen	0,0333	0,0714
ti14	kurang	0,0333	0,0714
ti15	modularitas	0,0333	0
ti16	mudah	0,1	0
ti17	mungkin	0,0333	0
ti18	pelihara	0,0333	0
ti19	pemrograman	0,0333	0,0714
ti20	penting	0,0333	0,0714
ti21	pisah	0,0333	0
ti22	proses	0	0,0714
ti23	sistem	0,0333	0
ti24	tingkat	0,0667	0,0714

Berdasarkan Tabel 15, diperoleh hasil perhitungan Term Frequency (TF) untuk setiap term pada dokumen jawaban referensi (D1) dan jawaban kandidat (D2). Nilai TF menunjukkan proporsi kemunculan suatu term terhadap jumlah seluruh term dalam dokumen setelah melalui tahapan preprocessing. Semakin sering suatu term muncul pada dokumen, maka semakin besar nilai TF yang dihasilkan. Nilai TF tersebut selanjutnya digunakan sebagai dasar dalam proses perhitungan *Inverse Document Frequency* dan pembobotan TF-IDF.

f. Document Frequency (DF)

Tahap *Document Frequency* (DF) bertujuan untuk menghitung jumlah dokumen yang mengandung setiap term hasil preprocessing. Berbeda dengan *Term Frequency* (TF) yang memperhatikan frekuensi kemunculan suatu term dalam satu dokumen, DF hanya menghitung apakah suatu term muncul atau tidak pada setiap dokumen. Oleh karena itu, kemunculan *term* lebih dari satu kali dalam dokumen yang sama tetap dihitung sebagai satu dokumen. Perhitungan DF dilakukan menggunakan Persamaan

$$DF(t) = \sum_{d=1}^N \begin{cases} 1, & \text{Jika term } t \text{ muncul pada dokumen } d \\ 0, & \text{Jika term } t \text{ tidak muncul pada dokumen } d \end{cases}$$

Keterangan:

- DF(t) = jumlah dokumen yang mengandung term t
- N = jumlah seluruh dokumen



© 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

Pada penelitian ini digunakan dua dokumen, yaitu jawaban referensi (D1) dan jawaban kandidat (D2). Sebagai contoh, term *abstraksi* hanya terdapat pada dokumen D1 dan tidak muncul pada dokumen D2, sehingga nilai DF dihitung sebagai berikut

$$DF(abstraksi) = 1$$

Sementara itu, term kode muncul pada dokumen D1 dan juga pada dokumen D2, sehingga nilai DF diperoleh sebagai berikut.

$$TF(kode) = 2$$

Hasil perhitungan DF untuk seluruh term ditunjukkan pada Tabel 16.

Tabel 16 Hasil Perhitungan DF

ti	Term	DF
ti1	abstraksi	1
ti2	antar	2
ti3	antarmuka	2
ti4	awat	1
ti5	bantu	2
ti6	gantung	2
ti7	guna	2
ti8	implementasi	1
ti9	kembali	2
ti10	kembang	1
ti11	kode	2
ti12	kompleks	1
ti13	komponen	2
ti14	kurang	2
ti15	modularitas	1
ti16	mudah	1
ti17	mungkin	1
ti18	pelihara	1
ti19	pemrograman	2
ti20	penting	2
ti21	pisah	1
ti22	proses	1
ti23	sistem	1
ti24	tingkat	2



© 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

Berdasarkan Tabel 16, sebagian term memiliki nilai *Document Frequency* (DF) sebesar 1, yang menunjukkan bahwa term tersebut hanya muncul pada salah satu dokumen. Sementara itu, beberapa term seperti antar, antarmuka, bantu, gantung, guna, kembali, kode, komponen, kurang, pemrograman, penting, dan tingkat memiliki nilai DF sebesar 2, yang menunjukkan bahwa term tersebut muncul pada kedua dokumen. Nilai DF selanjutnya digunakan untuk menghitung *Inverse Document Frequency* (IDF), di mana term yang muncul pada lebih sedikit dokumen akan memperoleh bobot IDF yang lebih besar, sedangkan term yang muncul pada lebih banyak dokumen akan memperoleh bobot IDF yang lebih kecil.

g. Inverse Document Frequency (IDF)

Tahap *Inverse Document Frequency* (IDF) digunakan untuk mengukur tingkat kelangkaan suatu term di dalam kumpulan dokumen. Berbeda dengan TF yang mengukur frekuensi kemunculan kata pada satu dokumen, semakin banyak dokumen yang mengandung suatu term, maka nilai bobot IDF term tersebut akan semakin kecil, karena term tersebut dianggap sebagai kata umum yang kurang mampu membedakan satu dokumen dengan dokumen lainnya. Formulasi matematis untuk menghitung nilai IDF mengacu pada persamaan berikut:

$$IDF(t) = \ln\left(\frac{N}{df_t}\right) + 1$$

Keterangan:

- $IDF(t) = \ln(N/DF(t)) + 1$, yaitu nilai Inverse Document Frequency untuk term t
- N = jumlah total dokumen dalam korpus (pada penelitian ini, $N = 2$, yaitu dokumen jawaban referensi dan dokumen jawaban kandidat)
- $DF(t)$ = jumlah dokumen yang mengandung term t
- $\ln$ = fungsi logaritma natural
- $+1$ = konstanta penyesuaian (smoothing) yang ditambahkan pada hasil perhitungan IDF untuk menghindari nilai bobot menjadi nol pada saat sebuah term muncul di seluruh dokumen dalam korpus ($N = DF(t)$)

Pada penelitian ini, korpus yang digunakan untuk setiap pertanyaan terdiri atas 2 dokumen, yaitu jawaban referensi (D1) dan jawaban kandidat (D2), sehingga $N = 2$. Berikut adalah dua contoh perhitungan yang mewakili dua kondisi berbeda



© 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

1. Term yang hanya muncul di satu dokumen (term unik)

Sebagai contoh term "abstraksi" hanya muncul pada dokumen D1 (jawaban referensi), dan tidak terdapat pada D2, sehingga $DF = 1$, sedangkan $N = 2$ (total dokumen D1 dan D2). Perhitungannya adalah sebagai berikut:

$$IDF(abstraksi) = \ln \left(\frac{2}{1} \right) + 1 = 0,6931 + 1 = 1.6931$$

Nilai IDF yang tinggi (1,6931) ini menunjukkan bahwa term "abstraksi" bersifat khas atau unik, karena hanya muncul pada salah satu dari dua dokumen, sehingga term ini memiliki daya pembeda yang tinggi antara jawaban referensi dan jawaban kandidat.

2. Term yang muncul di seluruh dokumen (term umum)

Term "kode" muncul baik pada dokumen D1 maupun D2, sehingga $DF = 2$, sama dengan $N = 2$. Perhitungannya adalah sebagai berikut:

$$IDF(kode) = \ln \left(\frac{2}{2} \right) + 1 = 0 + 1 = 1$$

Nilai IDF yang rendah (1) ini menunjukkan bahwa term "kode" merupakan kata umum yang muncul pada kedua dokumen, sehingga daya pembedanya lebih rendah dibandingkan term yang hanya muncul pada salah satu dokumen. Nilai IDF inilah yang kemudian akan dikalikan dengan nilai TF pada tahap pembobotan TF-IDF ($W = TF \times IDF$) untuk memperoleh vektor bobot akhir dari masing-masing dokumen. Berikut ini hasil perhitungan IDF untuk setiap term hasil preprocessing ditunjukkan pada Tabel 17.

Tabel 17 Hasil Perhitungan IDF

ti	Term	DF	N	IDF = $\ln(N/DF)+1$
ti1	abstraksi	1	2	1,6931
ti2	antar	2	2	1
ti3	antarmuka	2	2	1
ti4	awat	1	2	1,6931
ti5	bantu	2	2	1
ti6	gantung	2	2	1
ti7	guna	2	2	1
ti8	implementasi	1	2	1,6931
ti9	kembali	2	2	1
ti10	kembang	1	2	1,6931



© 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

ti11	kode	2	2	1
ti12	kompleks	1	2	1,6931
ti13	komponen	2	2	1
ti14	kurang	2	2	1
ti15	modularitas	1	2	1,6931
ti16	mudah	1	2	1,6931
ti17	mungkin	1	2	1,6931
ti18	pelihara	1	2	1,6931
ti19	pemrograman	2	2	1
ti20	penting	2	2	1
ti21	pisah	1	2	1,6931
ti22	proses	1	2	1,6931
ti23	sistem	1	2	1,6931
ti24	tingkat	2	2	1

h. Perhitungan Bobot Akhir TF-IDF (W)

Setelah nilai TF dan IDF untuk setiap term diperoleh, tahap selanjutnya adalah menghitung bobot TF-IDF. Bobot TF-IDF merupakan hasil perkalian antara nilai TF (yang menunjukkan seberapa sering suatu term muncul pada sebuah dokumen) dengan nilai IDF (yang menunjukkan seberapa khas term tersebut di dalam korpus). Semakin tinggi nilai bobot TF-IDF suatu term pada sebuah dokumen, maka semakin penting dan semakin merepresentasikan term tersebut terhadap dokumen itu. Formulasi matematis untuk menghitung nilai bobot TF-IDF mengacu pada persamaan berikut:

$$TF - IDF(t, d) = TF(t, d) \times IDF(t)$$

Keterangan:

- $TF-IDF(t, d)$ = bobot term t pada dokumen d
- $TF(t, d)$ = nilai Term Frequency term t pada dokumen d
- $IDF(t)$ = nilai Inverse Document Frequency term t

Pada penelitian ini, pembobotan TF-IDF dihitung untuk setiap term pada masing-masing dokumen, yaitu dokumen jawaban referensi (D1) dan dokumen jawaban kandidat (D2). Berikut adalah dua contoh perhitungan yang mewakili dua kondisi berbeda.



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. Term yang hanya muncul di satu dokumen (term unik)

Term "abstraksi" hanya muncul pada dokumen D1 dengan nilai $TF(abstraksi, D1) = 0,0333$, sedangkan pada D2 term ini tidak muncul sehingga $TF(abstraksi, D2) = 0$. Nilai $IDF(abstraksi) = 1,6931$ (dilihat Tabel 17). Perhitungan bobot TF-IDF-nya adalah sebagai berikut:

$$TF - IDF(abstraksi, D1) = 0,0333 \times 1,6931 = 0,0564$$

$$TF - IDF(abstraksi, D2) = 0 \times 1,6931 = 0$$

Bobot pada D1 bernilai 0,0564 karena term ini muncul pada dokumen referensi, sedangkan bobotnya pada D2 bernilai 0 karena term tersebut sama sekali tidak muncul pada jawaban kandidat.

2. Term yang muncul di seluruh dokumen (term umum)

Term "kode" muncul pada kedua dokumen, dengan nilai $TF(kode, D1) = 0,0667$ dan $TF(kode, D2) = 0,1$. Nilai $IDF(kode) = 1$ (dilihat Tabel 17). Perhitungan bobot TF-IDF-nya adalah sebagai berikut:

$$TF - IDF(kode, D1) = 0,0667 \times 1 = 0,0667$$

$$TF - IDF(kode, D2) = 0,0714 \times 1 = 0,0714$$

Karena nilai IDF term "kode" rendah (bernilai 1), maka bobot TF-IDF-nya pada kedua dokumen relatif kecil meskipun term ini cukup sering muncul, sebab term tersebut dianggap kurang memiliki daya pembeda antar dokumen. Nilai-nilai bobot TF-IDF inilah yang kemudian disusun menjadi vektor D1 dan vektor D2, untuk selanjutnya digunakan pada tahap perhitungan Cosine Similarity.

Berikut ini hasil perhitungan bobot TF-IDF untuk setiap term ditunjukkan pada Tabel 18.

Tabel 18 Hasil Perhitungan TF-IDF

ti	Term	TF (D1)	TF (D2)	IDF	TF-IDF (D1)	TF-IDF (D2)
ti1	abstraksi	0,0333	0	1,6931	0,0564	0
ti2	antar	0,0333	0,0714	1	0,0333	0,0714
ti3	antarmuka	0,1333	0,1429	1	0,1333	0,1429
ti4	awat	0,0333	0	1,6931	0,0564	0
ti5	bantu	0,0333	0,0714	1	0,0333	0,0714
ti6	gantung	0,0333	0,0714	1	0,0333	0,0714



© 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

ti7	guna	0,0333	0,0714	1	0,0333	0,0714
ti8	implementasi	0,0333	0	1,6931	0,0564	0
ti9	kembali	0,0333	0,0714	1	0,0333	0,0714
ti10	kembang	0,0333	0	1,6931	0,0564	0
ti11	kode	0,0667	0,0714	1	0,0667	0,0714
ti12	kompleks	0,0333	0	1,6931	0,0564	0
ti13	komponen	0,0333	0,0714	1	0,0333	0,0714
ti14	kurang	0,0333	0,0714	1	0,0333	0,0714
ti15	modularitas	0,0333	0	1,6931	0,0564	0
ti16	mudah	0,1	0	1,6931	0,1693	0
ti17	mungkin	0,0333	0	1,6931	0,0564	0
ti18	pelihara	0,0333	0	1,6931	0,0564	0
ti19	pemrograman	0,0333	0,0714	1	0,0333	0,0714
ti20	penting	0,0333	0,0714	1	0,0333	0,0714
ti21	pisah	0,0333	0	1,6931	0,0564	0
ti22	proses	0	0,0714	1,6931	0	0,1209
ti23	sistem	0,0333	0	1,6931	0,0564	0
ti24	tingkat	0,0667	0,0714	1	0,0667	0,0714

3. Perhitungan Nilai Cosine Similarity

Setelah diperoleh hasil pembobotan TF-IDF pada tahap sebelumnya (Tabel 18), tahap selanjutnya adalah menghitung tingkat kemiripan (*similarity*) antara vektor TF-IDF jawaban referensi (D1) dan vektor TF-IDF jawaban kandidat (D2) menggunakan metode *Cosine Similarity*. Metode ini mengukur kemiripan dua buah vektor berdasarkan sudut di antara keduanya dalam ruang vektor multidimensi (Syah et al., 2025). Nilai *cosine similarity* berkisar antara 0 (sama sekali tidak mirip) hingga 1 (sangat mirip). Formulasi matematis untuk menghitung nilai cosine similarity mengacu pada persamaan berikut:

$$\cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

Keterangan:

- $\cos(\theta)$ = nilai cosine similarity antara vektor A dan vektor B
- A = vektor TF-IDF dokumen referensi (D1)
- B = vektor TF-IDF dokumen kandidat (D2)



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

- $A_i B_i$ = bobot TF-IDF term ke-i pada vektor A dan vektor B
- $A.B$ = dot product (hasil kali titik) antara vektor A dan vektor B, dihitung dengan $\sum_{i=1}^n A_i B_i$
- $\|A\|$ = magnitude (panjang) vektor A, dihitung dengan $\sqrt{\sum_{i=1}^n A_i^2}$
- $\|B\|$ = magnitude (panjang) vektor B, dihitung dengan $\sqrt{\sum_{i=1}^n B_i^2}$
- n = jumlah term dalam vocabulary gabungan (n = 30)

Perhitungan dilakukan dengan mengalikan bobot TF-IDF antar term yang bersesuaian pada kedua vektor $A_i \times B_i$, kemudian menjumlahkan seluruh hasil perkalian tersebut untuk mendapatkan nilai dot product. Selanjutnya, dihitung pula magnitude (panjang) masing-masing vektor dengan menjumlahkan kuadrat setiap elemen vektor lalu diakarkan. Rincian perhitungan untuk setiap term ditunjukkan pada Tabel 19.

Tabel 19 Hasil Perhitungan Cosine Similarity

ti	Term	A_i (TF-IDF D1)	B_i (TF-IDF D2)	$A_i \times B_i$	A_i^2	B_i^2
ti1	abstraksi	0,0564	0	0	0,0032	0
ti2	antar	0,0333	0,0714	0,0024	0,0011	0,0051
ti3	antarmuka	0,1333	0,1429	0,0190	0,0178	0,0204
ti4	awat	0,0564	0	0	0,0032	0
ti5	bantu	0,0333	0,0714	0,0024	0,0011	0,0051
ti6	gantung	0,0333	0,0714	0,0024	0,0011	0,0051
ti7	guna	0,0333	0,0714	0,0024	0,0011	0,0051
ti8	implementasi	0,0564	0	0,0000	0,0032	0
ti9	kembali	0,0333	0,0714	0,0024	0,0011	0,0051
ti10	kembang	0,0564	0	0	0,0032	0
ti11	kode	0,0667	0,0714	0,0048	0,0044	0,0051
ti12	kompleks	0,0564	0	0	0,0032	0
ti13	komponen	0,0333	0,0714	0,0024	0,0011	0,0051
ti14	kurang	0,0333	0,0714	0,0024	0,0011	0,0051
ti15	modularitas	0,0564	0	0	0,0032	0
ti16	mudah	0,1693	0	0	0,0287	0
ti17	mungkin	0,0564	0	0	0,0032	0
ti18	pelihara	0,0564	0	0	0,0032	0



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

ti19	pemrograman	0,0333	0,0714	0,0024	0,0011	0,0051
ti20	penting	0,0333	0,0714	0,0024	0,0011	0,0051
ti21	pisah	0,0564	0	0	0,0032	0
ti22	proses	0	0,1209	0	0,0000	0,0146
ti23	sistem	0,0564	0	0	0,0032	0
ti24	tingkat	0,0667	0,0714	0,0048	0,0044	0,0051
$\sum$ (Jumlah)				0,0500	0,0971	0,0911

Berdasarkan Tabel 19, dapat dilihat bahwa dari 30 term dalam vocabulary, hanya term "kode" dan "komponen" yang memberikan kontribusi pada nilai $A_i \times B_i$, karena hanya kedua term tersebut yang muncul pada kedua dokumen (D1 dan D2) sekaligus. Sedangkan 28 term lainnya bersifat unik (hanya muncul pada salah satu dokumen), sehingga hasil perkaliannya bernilai 0 dan tidak memberikan kontribusi pada dot product.

Berdasarkan hasil nilai dot product pada Tabel 19, diperoleh:

$$\begin{aligned}
 A \cdot B &= \sum_{i=1}^{30} A_i B_i = 0,0024 + 0,0190 + 0,0024 + 0,0024 + 0,0024 + \\
 &0,0024 + 0,0048 + 0,0024 + 0,0024 + 0,0024 + 0,0024 + 0,0048 \\
 &= 0,0500
 \end{aligned}$$

Selain nilai dot product ($A \cdot B$), untuk menghitung cosine similarity juga diperlukan nilai magnitude (panjang) dari masing-masing vektor, yaitu $\|A\|$ untuk vektor D1 dan $\|B\|$ untuk vektor D2. Magnitude vektor A ($\|A\|$) diperoleh dengan cara mengkuadratkan setiap nilai A_i (kolom TF – IDF D1) pada Tabel 19, menjumlahkan seluruh hasil kuadrat tersebut, kemudian mengakarkannya (akar pangkat dua). Dengan cara yang sama, magnitude vektor B ($\|B\|$) diperoleh dengan mengkuadratkan setiap nilai B_i (kolom TF – IDF D2), menjumlahkan seluruhnya, lalu mengakarkannya. Term yang bernilai 0 pada suatu dokumen tidak memberikan kontribusi apa pun terhadap magnitude dokumen tersebut, karena $0^2 = 0$.

$$\|A\| = \sqrt{\sum_{i=1}^{30} A_i^2} = \sqrt{0,0971} = 0,3116$$



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

$$\| B \| = \sqrt{\sum_{i=1}^{30} B_i^2} = \sqrt{0,0911} = 0,3019$$

Nilai $\| A \|$ merepresentasikan panjang vektor TF-IDF dari jawaban referensi (D1), sedangkan Nilai $\| B \|$ merepresentasikan panjang vektor TF-IDF dari jawaban kandidat (D2). Sehingga nilai cosine similarity antara jawaban referensi (D1) dan jawaban kandidat (D2) yaitu

$$\cos(\theta) = \frac{A \cdot B}{\| A \| \| B \|} = \frac{0,0500}{0,3116 \times 0,3019} = \frac{0,0500}{0,0941} = 0,5312$$

Berdasarkan hasil perhitungan diperoleh nilai cosine similarity sebesar 0,5312. Nilai tersebut menunjukkan bahwa jawaban kandidat memiliki tingkat kemiripan sedang terhadap jawaban referensi. Hal ini terlihat dari adanya beberapa term yang muncul pada kedua dokumen, seperti antar, antarmuka, bantu, gantung, guna, kembali, kode, komponen, kurang, pemrograman, penting, dan tingkat. Namun, masih terdapat sejumlah term penting yang hanya muncul pada jawaban referensi, seperti abstraksi, implementasi, kompleks, modularitas, mudah,elihara, pisah, dan sistem. Kondisi tersebut menunjukkan bahwa meskipun jawaban kandidat telah membahas topik yang sama, yaitu konsep antarmuka, penjelasan yang diberikan belum mencakup seluruh konsep utama yang terdapat pada jawaban referensi. Sehingga, tingkat kemiripan yang dihasilkan berada pada kategori sedang dan belum menunjukkan kesamaan isi yang tinggi.

4.3.4 Implementasi Speech-to-Text Menggunakan Whisper

Implementasi *Speech-to-Text* pada sistem ini menggunakan model *Whisper* Large V3 untuk mengubah masukan suara pengguna menjadi representasi teks yang selanjutnya digunakan pada proses evaluasi jawaban dan pembentukan pertanyaan adaptif. Pemilihan *Whisper* dilakukan karena model ini memiliki kemampuan pengenalan ucapan yang baik pada berbagai kondisi audio serta memiliki ketahanan terhadap kebisingan (*noise-robust speech recognition*) tanpa memerlukan tahapan *noise filtering* yang kompleks. Selain itu, pendekatan ini dipilih untuk mempertahankan karakteristik sistem wawancara yang bersifat *real-time* sehingga proses transkripsi dapat dilakukan dengan latensi yang rendah.



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

Pada implementasinya, proses *Speech-to-Text* terdiri atas dua tahapan, yaitu implementasi akuisisi audio dan konfigurasi audio, serta implementasi transkripsi audio menggunakan *Whisper*.

1. Implementasi Akuisisi dan Konfigurasi Audio

Tahap akuisisi dan konfigurasi audio merupakan tahap awal dalam mekanisme *Speech-to-Text* yang bertujuan untuk memperoleh masukan suara dari pengguna sebelum dilakukan proses transkripsi. Tahap ini diimplementasikan pada sisi frontend menggunakan Web Media API melalui fungsi navigator.mediaDevices.getUserMedia() untuk mengakses perangkat mikrofon pada browser.

Pada tahap ini, sistem menerapkan beberapa parameter konfigurasi audio guna meningkatkan kualitas masukan suara, meliputi echo cancellation, automatic gain control, noise suppression, jumlah kanal audio (*channel count*), dan sample rate. Parameter-parameter tersebut berfungsi untuk meredam gangguan suara latar, menjaga kestabilan volume, serta menyesuaikan format audio agar sesuai dengan kebutuhan proses *Speech-to-Text* secara real-time. Penjelasan lebih lanjut mengenai masing-masing parameter akuisisi audio disajikan pada Tabel 20.

Tabel 20 Konfigurasi Parameter Akuisisi Audio

Parameter	Nilai	Deskripsi
echoCancellation	true	Mengurangi pantulan suara (echo) yang berasal dari perangkat audio selama proses perekaman
autoGainControl	true	Menjaga kestabilan intensitas suara agar volume masukan lebih konsisten
noiseSuppression	true	Mengurangi kebisingan latar (background noise) selama proses akuisisi audio
channelCount	1	Menggunakan format audio mono untuk mengurangi kompleksitas pemrosesan
sampleRate	16000	Menyesuaikan frekuensi pengambilan sampel audio untuk kebutuhan proses <i>Speech-to-Text</i>

Berdasarkan Tabel 20, sistem menerapkan konfigurasi audio bawaan browser sebagai bentuk preprocessing ringan sebelum proses transkripsi dilakukan. Konfigurasi tersebut dipilih untuk meningkatkan kualitas masukan suara tanpa menambah kompleksitas maupun waktu pemrosesan yang signifikan. Pendekatan



© 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

ini juga mendukung kebutuhan sistem wawancara yang bersifat *real-time*, sehingga proses transkripsi dapat berjalan dengan latensi yang tetap rendah tanpa memerlukan *filtering* audio tambahan pada tahap akuisisi.

2. Implementasi Transkripsi Menggunakan *Whisper*

Whisper Large V3 dimanfaatkan melalui API Groq untuk transkripsi berbasis cloud guna mempercepat waktu pemrosesan audio. Salah satu pertimbangan utama dalam konfigurasi model adalah penentuan parameter yang mampu menghasilkan transkripsi akurat dalam konteks percakapan teknis bidang teknologi informasi.

Untuk mendukung akurasi tersebut, diterapkan teknik prompt engineering dengan menyertakan konteks domain secara eksplisit, mencakup istilah-istilah teknis yang umum digunakan dalam wawancara kerja bidang teknologi informasi seperti React.js, Next.js, TypeScript, PostgreSQL, Docker, Kubernetes, REST API, dan sejenisnya. Penyertaan prompt ini bertujuan untuk mengarahkan model agar lebih tepat dalam mengenali dan mentranskripsi istilah teknis yang berpotensi ambigu secara akustik.

Selain prompt, parameter language ditetapkan dengan nilai "id" untuk membatasi proses transkripsi pada Bahasa Indonesia, sehingga model tidak melakukan deteksi bahasa secara otomatis yang dapat menurunkan akurasi pada masukan suara dengan aksen atau kosakata teknis berbahasa Inggris. Parameter temperature ditetapkan dengan nilai 0.0 agar model menghasilkan transkripsi yang deterministik dan konsisten pada setiap masukan audio.

Penentuan nilai ini dilakukan melalui uji coba terhadap beberapa variasi nilai, yaitu 0.5 dan 1.0. Hasil uji coba menunjukkan bahwa nilai 0.5 dan 1.0 menghasilkan variasi transkripsi yang tidak stabil, ditandai dengan munculnya kata-kata tambahan atau kesalahan transkripsi pada istilah teknis tertentu. Oleh karena itu, nilai 0.0 dipilih karena pada sistem wawancara *real-time*, konsistensi keluaran transkripsi lebih diutamakan dibandingkan variasi hasil prediksi. Pengaturan parameter lengkap pada model Whisper Large V3 disajikan pada Tabel 21.

Tabel 21 Paramater Whisper

Parameter	Nilai	Deskripsi
model	"whisper-large-v3"	Model whisper digunakan untuk transkripsi audio, mendukung pengenalan ucapan multibahasa.



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

prompt	“konteks wawancara kerja bidang teknologi informasi”	Instruksi khusus untuk memandu model agar fokus mengenali istilah teknis dalam konteks wawancara kerja bidang teknologi informasi
language	“id”	Menetapkan bahasa audio yang akan ditranskripsi sebagai Bahasa Indonesia
temperature	0.0	Nilai 0.0 dipilih agar transkripsi bersifat konsisten dan deterministik.

Setelah konfigurasi parameter ditetapkan, proses transkripsi dijalankan sebagai berikut. Audio yang diperoleh dari masukan suara pengguna disimpan secara temporer pada sisi backend, kemudian dikirimkan ke model Whisper Large V3 melalui API Groq beserta parameter konfigurasi sebagaimana disajikan pada Tabel 21. Model selanjutnya memproses audio tersebut dan menghasilkan keluaran berupa teks transkripsi yang merepresentasikan jawaban lisan pengguna. Hasil transkripsi ini kemudian diteruskan ke tahap berikutnya, yaitu proses evaluasi jawaban dan pembentukan pertanyaan adaptif oleh model bahasa.

4.3.5 Implementasi Antarmuka Pengguna

Tahap ini dirancang antarmuka aplikasi InterviewMate yang mengintegrasikan model RAG ke dalam proses simulasi wawancara. Pengguna dapat melakukan registrasi atau login, memilih kategori, role, dan level wawancara, kemudian mengikuti sesi wawancara interaktif dengan AI yang menghasilkan pertanyaan secara dinamis. Setelah sesi selesai, sistem menampilkan transkrip percakapan serta laporan yang berisi ringkasan substansi wawancara. Tampilan antarmuka aplikasi InterviewMate disajikan pada bagian berikut.



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. Antarmuka Pengguna Login

Gambar 4.32 Antarmuka Pengguna Halaman Login

Gambar 4.32 menampilkan halaman login sebagai gerbang awal untuk mengakses platform InterviewMate. Pengguna diminta memasukkan nama pengguna dan kata sandi yang telah terdaftar, dengan tersedianya opsi untuk menampilkan atau menyembunyikan kata sandi. Lalu bagi pengguna yang belum memiliki akun, disediakan tautan untuk melakukan pendaftaran. Halaman ini dirancang sederhana dan intuitif guna memudahkan proses autentikasi sekaligus menjaga keamanan data dan riwayat wawancara pengguna.

2. Antarmuka Pengguna Pendaftaran

Gambar 4.33 Antarmuka Pengguna Halaman Pendaftaran

Gambar 4.33 menampilkan halaman pendaftaran yang digunakan untuk membuat akun baru pada platform InterviewMate. Pengguna diminta mengisi nama



© 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

pengguna, alamat email, kata sandi, dan konfirmasi kata sandi, dengan tersedianya opsi untuk menampilkan atau menyembunyikan kata sandi. Setelah seluruh data diisi dengan benar, pengguna dapat menekan tombol daftar untuk menyelesaikan proses registrasi. Lalu bagi pengguna yang sudah memiliki akun, disediakan tautan untuk kembali ke halaman login.

3. Antarmuka Pengguna Interview Room

Gambar 4.34 Antarmuka Pengguna Halaman Interview Room

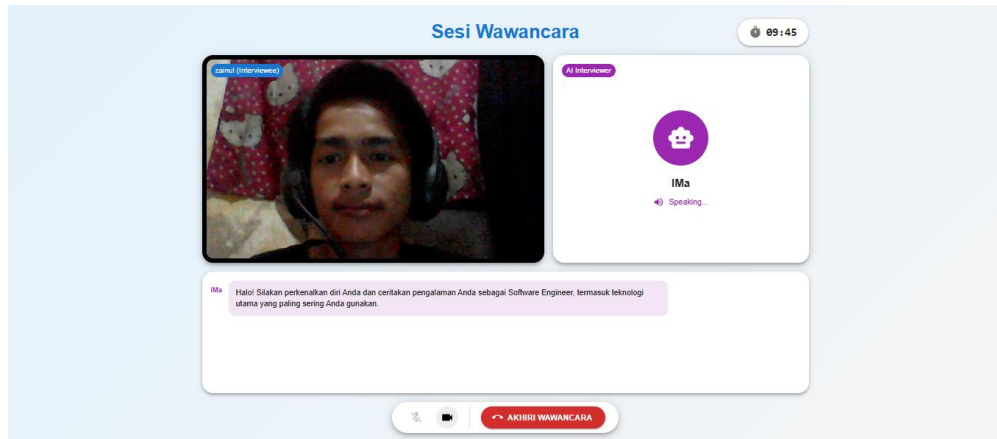
Gambar 4.34 menampilkan halaman interview room yang berfungsi sebagai tahap awal sebelum pengguna memulai sesi wawancara. Pada halaman ini, pengguna diminta memilih kategori bidang, role, dan level kemampuan yang diinginkan. Pilihan tersebut digunakan oleh sistem untuk menyesuaikan konteks dan tingkat kesulitan pertanyaan melalui model RAG. Setelah semua pilihan diisi, pengguna dapat menekan tombol "Mulai Wawancara" untuk melanjutkan ke sesi wawancara interaktif.



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

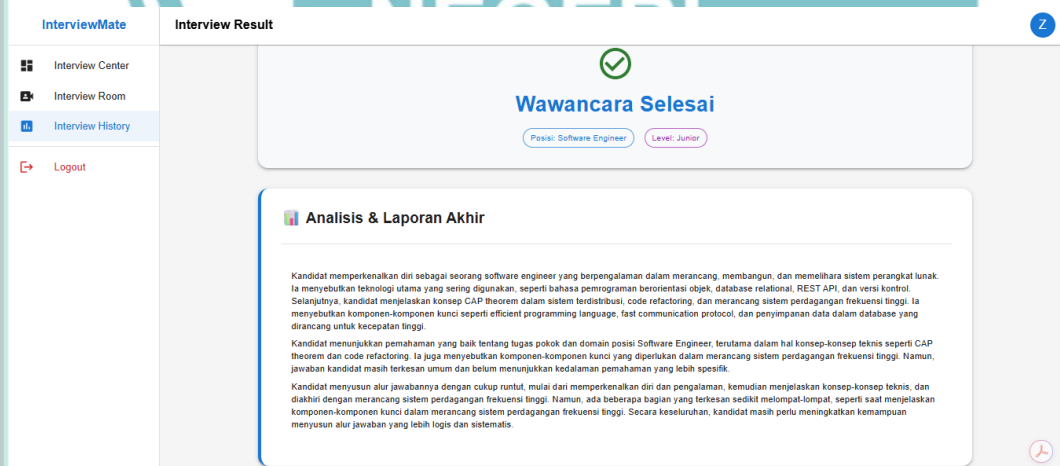
4. Antarmuka Pengguna Halaman Sesi Wawancara



Gambar 4.35 Antarmuka Pengguna Halaman Sesi Wawancara

Gambar 4.35 menampilkan halaman sesi wawancara yang menjadi inti dari platform InterviewMate. Pada halaman ini, pengguna melakukan simulasi wawancara secara langsung dengan Ima AI (InterviewMate AI) yang dilengkapi tampilan video pengguna dan identitas AI untuk memberikan pengalaman yang menyerupai wawancara nyata. Pertanyaan ditampilkan secara bertahap pada area percakapan dan dihasilkan berdasarkan konteks menggunakan model RAG. Selain itu, tersedia indikator waktu untuk menunjukkan durasi wawancara, serta kontrol mikrofon, kamera, dan tombol untuk mengakhiri sesi di bagian bawah halaman.

5. Antarmuka Pengguna Halaman Laporan



Gambar 4.36 Antarmuka Pengguna Halaman Laporan

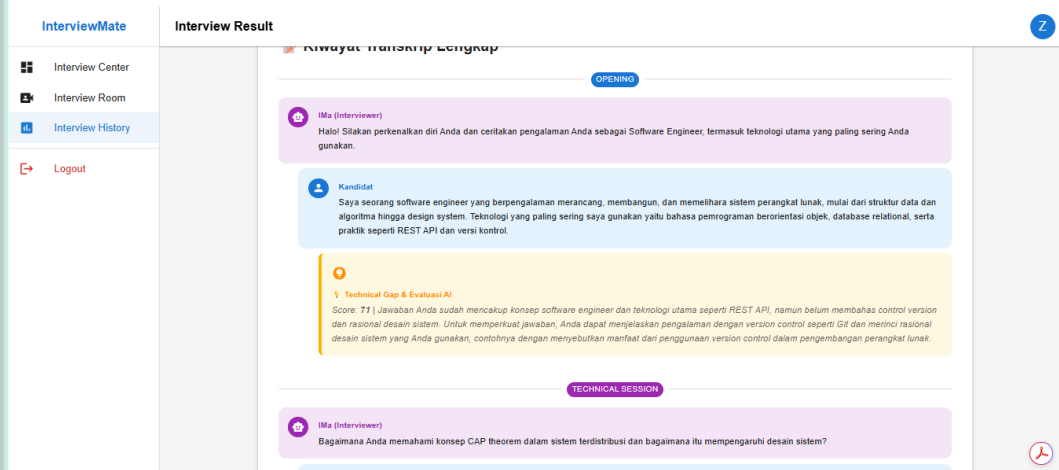
Gambar 4.36 menampilkan halaman laporan yang menyajikan hasil evaluasi dari sesi wawancara yang telah dilakukan. Halaman ini menampilkan analisis terstruktur



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

yang mencakup ringkasan, kekuatan teknis, kelemahan, dan rekomendasi pembelajaran yang dihasilkan secara otomatis oleh sistem berdasarkan jawaban pengguna selama sesi wawancara berlangsung.



Gambar 4.37 Antarmuka Pengguna Halaman Laporan Riwayat Transkripsi

Gambar 4.37 menampilkan riwayat transkrip lengkap dari sesi wawancara yang telah dilakukan. Pada bagian ini, seluruh percakapan antara Ima AI (InterviewMate AI) dan kandidat ditampilkan secara berurutan mulai dari tahap pembukaan hingga sesi teknis, disertai anotasi evaluasi pada setiap jawaban pengguna yang mencakup skor dan catatan perbaikan. Transkrip ditampilkan secara terstruktur dengan pemisahan antar sesi sehingga pengguna dapat meninjau kembali jawaban yang telah diberikan.

4.4 Pengujian

Tahap pengujian ini bertujuan untuk memastikan bahwa aplikasi yang dibuat sudah berjalan dengan baik dan sesuai dengan rancangan awal. Pengujian ini juga dilakukan untuk memvalidasi kinerja sistem, mengevaluasi hasil analisis model, dan melihat sejauh mana fitur-fitur aplikasi dapat berfungsi. Nantinya, pembahasan pada tahap ini akan diuraikan menjadi deskripsi pengujian, prosedur pengujian, serta hasil dan analisis data pengujian.

4.4.1 Deskripsi Pengujian

Tahap ini menguraikan batasan, fokus, serta skenario evaluasi yang dilakukan terhadap aplikasi. Untuk memastikan sistem berjalan dengan optimal dan sesuai dengan spesifikasi yang diharapkan, pelaksanaan evaluasi dibagi menjadi empat jenis pengujian utama.



© 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

1. Pengujian Kualitas Sistem RAG Menggunakan RAGAs

Pengujian kualitas sistem wawancara difokuskan pada fitur *generate* pertanyaan untuk mengevaluasi kinerja arsitektur *Retrieval-Augmented Generation* (RAG). Pengujian ini bertujuan memastikan sistem mampu mengambil konteks yang relevan dari *knowledge base* sehingga menghasilkan pertanyaan yang sesuai dengan posisi pekerjaan yang dipilih. Evaluasi dilakukan menggunakan framework RAGAs (*Retrieval-Augmented Generation Assessment*) untuk mengukur performa komponen *retrieval* dan *generation* secara objektif.

2. Pengujian Validitas Feedback Sistem dengan Metode Human Evaluation

Pengujian ini dilakukan untuk memvalidasi kualitas *feedback* yang dihasilkan sistem melalui penilaian dua orang pakar di bidang IT, yaitu seorang IT Senior dan seorang Manager. Evaluasi dilakukan menggunakan Google Sheet yang berisi 15 sampel berupa pertanyaan wawancara, jawaban kandidat, dan *feedback* sistem, yang kemudian dinilai secara independen oleh kedua pakar.

3. Pengujian Fungsionalitas Sistem dengan Metode Black Box Testing

Pengujian fungsionalitas sistem menggunakan metode *Black Box Testing* dilakukan untuk memastikan fitur-fitur aplikasi berjalan dengan baik tanpa perlu melihat bagaimana struktur kode di baliknya. Pada pengujian ini, fokus utamanya ada pada interaksi penggunaan. Sistem akan diberikan masukan tertentu, lalu dicek apakah hasil yang dikeluarkan sudah benar dan sesuai harapan. Pengujian ini dilakukan secara menyeluruh untuk memastikan tidak ada fungsi yang *error* dan sistem bekerja sesuai dengan spesifikasi yang telah ditetapkan.

4. Pengujian Usabilitas Sistem Menggunakan System Usability Scale (SUS)

Pengujian usabilitas sistem menggunakan kuesioner *System Usability Scale* (SUS) bertujuan untuk mengevaluasi tingkat kegunaan dan kenyamanan aplikasi dari sudut pandang pengguna. Dalam pengujian ini, pengguna akan memberikan penilaian setelah mencoba langsung fitur-fitur yang ada. Hasil dari pengukuran SUS ini akan menunjukkan apakah antarmuka sistem sudah mudah dipahami atau masih memerlukan perbaikan agar interaksinya lebih optimal.

5. Pengujian Kepuasan dan Loyalitas Pengguna dengan Net Promoter Score (NPS)

Pengujian menggunakan metode *Net Promoter Score* (NPS) dilakukan untuk mengevaluasi tingkat kepuasan sekaligus loyalitas pengguna setelah berinteraksi



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

dengan aplikasi InterviewMate. Metrik ini secara spesifik mengukur seberapa besar kecenderungan atau kesediaan pengguna untuk merekomendasikan sistem simulasi wawancara ini kepada orang lain.

4.4.2 Prosedur Pengujian

Tahap ini menjelaskan prosedur pengujian sistem yang dilakukan secara sistematis dan terstruktur. Pembahasan disusun berdasarkan lima skenario pengujian, yaitu evaluasi metrik RAG menggunakan RAGAs, validasi umpan balik sistem melalui *Human Evaluation*, pengujian fungsionalitas menggunakan Black Box Testing, serta evaluasi pengguna melalui *System Usability Scale (SUS)* dan *Net Promoter Score (NPS)*.

1. Prosedur Pengujian RAGAs

Prosedur pengujian ini dilakukan untuk mengevaluasi kualitas sistem RAG dengan mengukur tiga metrik utama, yaitu context precision, faithfulness, dan answer relevance. Pengujian ini dilakukan secara otomatis menggunakan skrip pemrograman dengan pendekatan LLM-as-a-Judge, di mana model AI digunakan sebagai penilai objektif. Urutan langkah pengujiannya adalah sebagai berikut:

a. Penyiapan Data Uji

Sistem membaca dan memuat data riwayat simulasi yang sebelumnya telah disimpan dalam format dokumen CSV. Data ini berisi riwayat pertanyaan AI, konteks yang ditarik dari database, dan posisi yang dilamar.

b. Proses Penilaian

Sistem melakukan perulangan pada setiap baris data pertanyaan untuk dinilai oleh AI *evaluator*. Pada tahap ini, AI akan memberikan skor pada skala 0.0 hingga 1.0 untuk ketiga metrik yang diuji.

c. Perhitungan Rata-Rata

Setelah seluruh pertanyaan dinilai, sistem menyingkirkan data yang kosong, lalu menghitung nilai rata-rata keseluruhan dari masing-masing metrik evaluasi.

2. Prosedur Pengujian Human Evaluation

Pengujian dengan metode *Human Evaluation* ini melibatkan dua orang pakar. Setiap pakar diminta mengisi lembar penilaian yang berisi 15 sampel data, dengan setiap sampel terdiri atas pertanyaan wawancara, jawaban kandidat, dan respon



© 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

yang dihasilkan sistem. Pertanyaan-pertanyaan tersebut diambil dari berbagai role/posisi pekerjaan yang berbeda.

Seluruh sampel diperoleh dari sesi wawancara yang dijalankan secara manual oleh peneliti. Selanjutnya, kedua pakar menilai setiap sampel jawaban tersebut ke dalam dua kategori, yaitu *valid* atau *invalid*.

3. Prosedur Pengujian Black Box Testing

Pengujian black box testing dilakukan untuk mengevaluasi kelancaran fungsionalitas sistem. Proses ini dijalankan dengan cara membuat skenario pengujian (*test case*) yang mencakup skenario keberhasilan (positif) maupun skenario kegagalan (negatif). Saat diuji, sistem akan dievaluasi dengan membandingkan antara kondisi yang seharusnya terjadi (*expected condition*) dengan respons nyata yang diberikan oleh sistem (*actual result*). Berdasarkan perbandingan tersebut, penguji akan menetapkan status keberhasilan untuk masing-masing fitur, yaitu: sesuai (*passed*), gagal (*failed*), atau tidak dapat dijalankan (*blocked*). Format pengisian dan hasil lengkap dari pengujian ini dapat dilihat pada contoh di Tabel 22.

Tabel 22 Prosedur Pengujian Black Box Testing

Test Case ID	Test Case Name	Test Condition	Test Case Desc	Expected Result	Actual Result	Test Case Status
TC_POS_Login_1	Verifikasi fungsionalitas login dengan menggunakan akun valid	Positif	Pengguna dapat masuk ke halaman utama dengan menggunakan username dan password yang telah didaftarkan	Pengguna masuk ke halaman utama	Pengguna masuk ke halaman utama	Sesuai

4. Prosedur Pengujian System Usability Scale (SUS)

Prosedur pengujian *System Usability Scale* (SUS) dilakukan untuk mengukur tingkat penerimaan dan kenyamanan pengguna terhadap antarmuka aplikasi. Pengujian ini dilakukan dengan melibatkan 10 orang responden yang mewakili



© 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

target pengguna sistem, Urutan langkah dan teknis pelaksanaannya adalah sebagai berikut:

a. Persiapan Kuesioner

Penguji menyiapkan instrumen kuesioner digital menggunakan platform Google Form. Kuesioner ini menggunakan 10 pertanyaan standar SUS (yang terbagi menjadi pernyataan positif dan negatif) dengan rincian seperti pada Tabel 23.

b. Sistem Penilaian

Setiap pertanyaan diukur menggunakan skala 1 sampai 5, di mana nilai 1 berarti "Sangat Tidak Setuju" dan nilai 5 berarti "Sangat Setuju".

c. Pelaksanaan Uji

Responden diinstruksikan untuk berinteraksi dengan aplikasi terlebih dahulu, lalu mengisi kuesioner secara objektif melalui tautan Google Form yang diberikan.

d. Pengolahan Skor

Data hasil kuesioner kemudian direkapitulasi dan dihitung menggunakan rumus standar SUS untuk mendapatkan skor akhir tingkat usability aplikasi.

Tabel 23 Persiapan Kuesioner SUS

Kode	Pertanyaan	1	2	3	4	5
SUS-1	Saya berpikir akan menggunakan sistem ini lagi					
SUS-2	Saya merasa sistem ini rumit untuk digunakan					
SUS-3	Saya merasa sistem ini mudah digunakan					
SUS-4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini					
SUS-5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya					
SUS-6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini)					
SUS-7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat					
SUS-8	Saya merasa sistem ini membingungkan					
SUS-9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini					
SUS-10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini					



© 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

5. Prosedur Pengujian Net Promoter Score (NPS)

Prosedur pengujian *Net Promoter Score* (NPS) dilakukan untuk mengevaluasi tingkat loyalitas pengguna akhir serta mengukur seberapa besar kesediaan mereka untuk merekomendasikan aplikasi InterviewMate kepada orang lain. Pengujian ini dilaksanakan segera setelah pengguna selesai melakukan simulasi wawancara, dengan melibatkan 10 orang responden yang secara demografi mewakili target utama pengguna sistem. Urutan langkah dan teknis pelaksanaan pengujian ini dijabarkan sebagai berikut:

a. Persiapan Instrumen Pengujian

Sistem menyiapkan formulir evaluasi yang memuat satu pertanyaan utama berstandar NPS. Pertanyaan ini disajikan menggunakan skala numerik dari 0 hingga 10, di mana nilai 0 merepresentasikan ketiadaan minat untuk merekomendasikan, dan nilai 10 merepresentasikan antusiasme yang sangat tinggi. Instrumen pertanyaan tersebut dapat dilihat pada Tabel 24.

b. Pengumpulan Data Responden

Kesepuluh responden yang telah menguji coba seluruh fungsionalitas sistem diminta untuk mengisi kuesioner NPS tersebut secara objektif berdasarkan pengalaman langsung yang mereka rasakan.

c. Klasifikasi Kategori Responden

Setelah seluruh data terkumpul, sistem atau peneliti melakukan rekapitulasi dan memecah jawaban responden ke dalam tiga kategori baku NPS. Responden yang memberikan skor 9–10 diklasifikasikan sebagai *Promoters*, skor 7–8 diklasifikasikan sebagai *Passives*, dan skor 0–6 diklasifikasikan sebagai *Detractors*.

d. Perhitungan Skor Akhir

Tahap terakhir adalah melakukan kalkulasi metrik NPS. Perhitungan dilakukan dengan mengabaikan persentase responden pasif (*Passives*), lalu mengurangi persentase total responden *Promoters* dengan persentase total responden *Detractors*. Hasil persentase akhir inilah yang akan menjadi tolok ukur kesuksesan penerimaan 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 mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Tabel 24 Pertanyaan NPS

Kode	Instrument Pertanyaan	Skala Penilaian
NPS-01	Seberapa besar kemungkinan Anda akan merekomendasikan aplikasi InterviewMate kepada teman, kerabat Anda?	0-10

4.4.3 Data Hasil Pengujian

Tahap ini menyajikan data hasil pengujian sistem untuk menggambarkan kinerja, fungsionalitas, dan tingkat penerimaan pengguna terhadap aplikasi. Pembahasan dilakukan berdasarkan lima skenario pengujian, yaitu evaluasi performa RAG menggunakan RAGAs, validitas umpan balik melalui *Human Evaluation*, pengujian fungsionalitas menggunakan *Black Box Testing*, serta pengujian *System Usability Scale (SUS)* dan *Net Promoter Score (NPS)*.

1. Pengujian RAGAs

Evaluasi sistem menggunakan framework RAGAs dilakukan terhadap riwayat simulasi wawancara yang telah terekam selama penggunaan sistem. Evaluasi menggunakan pendekatan LLM sebagai penilai untuk mengukur tiga metrik utama, yaitu *Context Precision*, *Faithfulness*, dan *Answer Relevance*. Hasil evaluasi dari ketiga metrik tersebut disajikan pada Tabel 25.

Tabel 25 Hasil Pengujian Evaluasi RAGAs

Metrik	Skor
Answer Relevance	0.9854
Context Precision	0.9200
Faithfulness	0.9413

Berdasarkan Tabel 25, hasil evaluasi sistem RAG menunjukkan performa yang sangat baik pada tiga metrik utama. Metrik *Answer Relevance* memperoleh skor 0,9854, yang menunjukkan bahwa respons AI sangat relevan dengan konteks wawancara. Metrik *Context Precision* memperoleh skor 0,9200, yang menandakan kemampuan sistem dalam mengambil konteks yang paling relevan dari *knowledge base*. Sementara itu, metrik *Faithfulness* memperoleh skor 0,9413, yang menunjukkan bahwa respons AI tetap konsisten dengan dokumen referensi sehingga mampu meminimalkan terjadinya halusinasi.



© 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

2. Hasil Pengujian Validitas Feedback sistem dengan Human Evaluation

Pengujian dengan metode human evaluation ini melibatkan dua orang pakar, yaitu seorang IT Senior dan seorang Manager, yang memiliki kompetensi dalam menilai kesesuaian antara jawaban kandidat, jawaban referensi, dan feedback yang dihasilkan oleh sistem. Kedua pakar diminta menilai 15 sampel data, yang masing-masing terdiri atas pertanyaan wawancara, jawaban kandidat, jawaban referensi, dan feedback yang dihasilkan sistem.

Tabel 26 Hasil Pengujian Human Evaluation

Pertanyaan	Jawaban Kandidat	Feedback	Pakar 1	Pakar 2
Jelaskan konsep Teorema CAP dalam sistem terdistribusi dan bagaimana ini mempengaruhi desain sistem?	Theorema CHAP menyatakan bahwa dalam sistem terdistribusi, kita tidak dapat secara bersamaan menjamin konsistensi, ketersediaan, dan toleransi partisi di saat yang sama. Hanya 2 dari 3 yang bisa dijamin penuh ketika terjadi partisi jaringan. Dalam desain sistem, ini berarti saya harus memilih trade-off. Misalnya, sistem finansial cenderung mengutamakan konsistensi dan toleransi partisi.	Konsep Teorema CAP yang sudah dijelaskan sudah cukup baik, namun masih kurang penekanan pada konsep ""tiga"" yang merupakan komponen utama teorema ini, yaitu consistency, availability, dan partition tolerance. Untuk memperkuat jawaban, kandidat dapat memberikan contoh spesifik tentang bagaimana trade-off antara ketiga komponen ini mempengaruhi desain sistem	Valid	Valid



© 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

	Sementara itu, sistem seperti media sosial lebih mengutamakan ketersediaan dan toleransi partisi.	terdistribusi, serta memberikan penjelasan yang lebih rinci tentang dampak dari setiap pilihan desain pada kinerja dan keandalan sistem.		
Bagaimana Anda dapat menerapkan prinsip abstraksi dalam pemrograman berorientasi objek untuk meningkatkan kemampuan kode Anda dalam menangani kompleksitas?	Untuk penerapan pada prinsip abstraksi yaitu menangani kompleksitas dengan cara menyembunyikan detail implementasi teknis yang rumit di balik sebuah antarmuka atau kelas abstrak yang sederhana. Pengembang hanya perlu berinteraksi dengan fungsi penting yang di-export ke publik tanpa harus tahu bagaimana fungsi tersebut bekerja di dalam sistem.	Untuk meningkatkan kemampuan kode dalam menangani kompleksitas dengan prinsip abstraksi, perlu ditekankan pada konsep "logika bisnis" dan "komplikasi" yang kurang dalam jawaban Anda. Saran untuk memperkuat jawaban adalah dengan memberikan contoh spesifik tentang bagaimana abstraksi dapat memisahkan logika bisnis dari detail teknis, sehingga	Valid	Valid



© 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

		mengurangi komplikasi dan meningkatkan kemudahan pemeliharaan kode.		
Apa itu overfitting dalam konteks machine learning, dan bagaimana cara mengatasinya dengan menggunakan teknik regularization?	Overfitting merupakan kondisi saat model terlalu menghapal data pelatihan hingga gagal memprediksi data baru dengan akurat. Untuk mengatasinya, Anda bisa menggunakan regularisasi yang menambahkan penalti pada model agar tidak terlalu kompleks, seperti L1 yaitu lasso yang dimana menghilangkan fitur yang tidak relevan dengan membuat bobotnya menjadi nol, Sedangkan L2 atau RIDGE ini mengecilkan bobot filter agar pengaruhnya lebih stabil dan tidak	Jawaban Anda sudah mencakup overfitting, namun belum membahas regularization dan loss secara menyeluruh. Untuk memperkuat jawaban, Anda dapat menjelaskan teknik regularization seperti L1 dan L2 regression dengan contoh singkat tentang bagaimana alpha mempengaruhi hasil model.	Invalid	Invalid



© 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

	ekstrim. Teknik ini memaksa model lebih fokus pada pola umum, bukan detail noise pada data latihan.			
Apa itu konsep bias-variance tradeoff dalam machine learning, dan bagaimana Anda menerapkan konsep ini untuk memilih model yang tepat untuk sebuah masalah?	Dalam konsep tersebut, yaitu bias variance trade-off merupakan sebuah keseimbangan antara kesalahan akibat model yang terlalu sederhana atau bias, atau bisa dikenal dengan underfitting, Yang dimana kesalahan akibat model yang terlalu kompleks sehingga menangkap noise seperti variance atau overfitting. Lalu untuk mengatasi tersebut, Anda menerapkannya dengan mengevaluasi performa model pada training dan validation data. Jika error tinggi,	Jawaban Anda sudah mencakup konsep bias dan variance, namun belum membahas tradeoff dan cross-validation secara mendalam. Untuk memperkuat jawaban, Anda dapat menjelaskan bagaimana konsep tradeoff antara bias dan variance dapat dioptimalkan dengan menggunakan teknik cross-validation dan regularisasi, seperti mencontohkan penggunaan k-fold cross-validation untuk mengevaluasi performa model.	Valid	Valid



© 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

	tingkatkan keremitan model, yaitu kurangi bias. Jika kurasi training sempurna tetapi error validasi tinggi, itu disederhanakan model atau gunakan regularization untuk mencari tingkat, untuk mencari titik keseimbangan terbaik.			
Apa yang dimaksud dengan pemfaktoran ulang kode dan mengapa praktik ini sangat penting dalam pengembangan perangkat lunak untuk meningkatkan kualitas dan kemudahan pemeliharaan kode yang ada	Pemfaktoran ulang Kode merupakan sebuah praktik meningkatkan struktur kode dan keterbacaan tanpa mengubah fungsionalitas program. Praktik ini penting karena dapat meningkatkan kemudahan pemeliharaan dan kualitas kode secara keseluruhan.	Jawaban Anda sudah mencakup pemfaktoran ulang kode, namun belum membahas keterbacaan kode dan refactoring. Untuk memperkuat jawaban, perlu menjelaskan bagaimana pemfaktoran ulang dapat meningkatkan keterbacaan kode dan memberikan contoh refactoring yang baik, seperti memisahkan kode	Invalid	Invalid



© 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

		menjadi fungsi-fungsi kecil yang lebih mudah dipahami.		
Bagaimana Anda memperbaiki kinerja sebuah query SQL yang kompleks?	Untuk memperbaiki kinerja query SQL yang kompleks, ada beberapa pendekatan utama. Yang pertama, optimalkan penggunaan index. Kedua, menulis SQL yang efisien. Ketiga, batasi pengambilan data. Keempat, analisis kinerja query. Dan yang kelima, pertimbangkan denormalisasi.	Jawaban Anda sudah mencakup query dan denormalisasi, namun belum membahas analisis kinerja query dan pembatasan jumlah data yang diambil. Untuk memperkuat jawaban, Anda dapat menjelaskan contoh konkret seperti menggunakan EXPLAIN untuk menganalisis rencana eksekusi query dan membatasi jumlah data yang diambil dengan menggunakan LIMIT atau pagination.	Invalid	Invalid
Bagaimana Anda akan	Keuntungan ORM menyederhanakan interaksi basis Data	Jawaban Anda sudah mencakup ORM dan interaksi	Invalid	Invalid



© 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

mengimplemen- ntasikan otentikasi dan otorisasi pada RESTful API untuk memastikan keamanan data pengguna?	dalam Menulis Kode bukan RAW SQL dan tidak bergantung pada bahasa atau database tertentu. Kerugiannya itu bisa menimbulkan overheat kinerja dan membatasi fitur database, kompleks atau spesifik, jadi sulit diakses.	basis data, namun belum membahas overhead kinerja dan optimasi. Untuk memperkuat jawaban, Anda dapat membahas contoh konkret tentang bagaimana mengimbangkan kemudahan penggunaan ORM dengan potensi overhead kinerja, seperti menggunakan teknik caching atau memilih ORM yang sesuai dengan kebutuhan proyek.		
Apa yang dimaksud dengan denormalisasi database dan kapan penggunaannya paling tepat?	Denormalisasi merupakan proses menambahkan redundancy data demi performa. Paling tepat dipakai pada skenario banyak baca atau jarang tulis, dengan trade-off mengorbankan penyimpanan dan kompleksitas dijaga	Jawaban Anda sudah mencakup denormalisasi database, namun belum membahas redundansi data dan konsep teknis lainnya seperti kompleksitas data. Untuk memperkuat jawaban, perlu menjelaskan	Invalid	Invalid



© 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

	konsekuensinya atau dijaga dengan konsistensi.	konsep redundansi data dan kompleksitas data secara lebih rinci, seperti bagaimana denormalisasi memperkenalkan redundansi data untuk meningkatkan kinerja dalam skenario tertentu.		
Apa itu prinsip desain 'Hierarchy' dalam UX Design dan berikan contoh penerapannya dalam sebuah aplikasi mobile?	Prinsip desain hirarki dalam UX Design adalah teknik untuk mengorganisir dan mengstruktur konten serta elemen desain dalam sebuah antarmuka pengguna, sehingga memandu perhatian pengguna ke informasi yang paling penting. Dalam sebuah aplikasi mobile, contoh penerapan hirarki dapat dilihat pada halaman utama aplikasi e-commerce. Misalnya, judul produk terbaru	Jawaban Anda sudah mencakup prinsip desain untuk mengorganisir dan mengstruktur konten serta elemen desain, namun belum membahas konsep hierarchy secara spesifik dan teknis. Untuk memperkuat jawaban, Anda dapat menjelaskan lebih lanjut tentang prinsip desain hierarchy dengan memberikan contoh seperti penggunaan ukuran font, warna,	Valid	Valid



© 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

	ditampilkan dengan font besar dan tebal, kemudian diikuti dengan gambar produk dan harga yang lebih kecil, dan selanjutnya, detail produk seperti deskripsi yang ulasannya ditampilkan dengan font yang lebih kecil lagi.	dan penempatan elemen untuk memandu perhatian pengguna ke informasi yang paling penting dalam sebuah aplikasi mobile.		
Bagaimana Anda menerapkan konsep 'Usability Testing' dalam proses desain UX untuk memvalidasi hipotesis desain dan meningkatkan kualitas pengalaman pengguna? Berikan contoh penerapan metode	Dalam menerapkan konsep usability testing, saya memulai dengan menentukan tujuan dan sasaran dari pengujian, seperti memahami bagaimana pengguna berintegrasi dengan aplikasi atau mengidentifikasi masalah dalam alur pengguna. Kemudian saya merekrut partisipan yang sesuai dengan persona target aplikasi.	Jawaban Anda sudah mencakup konsep usability testing, namun belum membahas secara mendalam tentang metode pengujian seperti A/B testing dan teknik analisis data pengguna. Untuk memperkuat jawaban, Anda dapat menjelaskan contoh penerapan metode usability testing dengan lebih spesifik, seperti melakukan	Valid	Valid



© 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

usability testing dalam sebuah proyek desain aplikasi mobile yang Anda ketahui.	menggunakan metode usability testing seperti wawancara, survei dan pengujian dengan prototipe atau aplikasi yang sudah jadi. Dalam pengujian, saya meminta partisipan untuk melakukan tugas-tugas spesifik dan merekam perilaku serta umpan balik mereka. Contohnya, dalam proyek desain aplikasi mobile, untuk memesan makanan, saya melakukan Usability Testing testing dengan meminta pengguna	pengujian dengan prototipe atau aplikasi yang sudah jadi dan menganalisis hasilnya untuk memvalidasi hipotesis desain.		
Bagaimana Anda menerapkan konsep 'Information Architecture' dalam desain	Dalam menerapkan konsep tersebut, saya menggunakan beberapa prinsip seperti membuat struktur navigasi yang jelas, menggunakan label	Jawaban Anda sudah mencakup konsep Information Architecture, namun belum membahas secara mendalam tentang teknik seperti card	Valid	Valid



© 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

UX untuk mengorganisir dan mempresentasikan konten secara efektif, sehingga memudahkan pengguna menemukan informasi yang dibutuhkan? Berikan contoh penerapan nyata dari sebuah aplikasi web atau mobile yang berhasil menerapkan prinsip Information Architecture ini.	dan kategori yang konsisten, serta memastikan bahwa konten disajikan dalam bentuk yang mulai dipahami. Contoh penerapan nyata dari konsep ini dapat dilihat pada aplikasi e-commerce seperti Lazada, dimana mereka menggunakan struktur kategori yang jelas dan sistem pencarian yang efektif untuk menggunakan pengguna menemukan produk yang dibutuhkan selain itu, mereka juga menggunakan teknik seperti face search dan filtering untuk memecah proses pencarian. Dalam proses desain, saya juga menggunakan teknik seperti consulting dan tree testing untuk	sorting dan wireframing yang dapat memperkuat struktur informasi. Untuk memperkuat jawaban, Anda dapat menjelaskan lebih lanjut tentang penerapan teknik-teknik tersebut, seperti menggunakan card sorting untuk memvalidasi struktur kategori dan wireframing untuk memvisualisasikan alur navigasi pengguna.		
---	--	--	--	--



© 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

	memvalidasi struktur informasi dan memastikan bahwa pengguna dapat menemukan informasi yang dibutuhkan dengan mudah.			
Apa kelebihan dari menggunakan arsitektur microservices dalam pengembangan sistem?	Kelebihan dalam Menggunakan Arsitektur Microservice Dalam Pengembangan Sistem Yaitu Arsitektur Ini Jenderung Untuk Project Yang Skalabilitasnya Itu Besar Yang dimana proses alur dari pengembangan ini, alur dari sistem yaitu asynchronous	Jawaban Anda belum membahas fleksibilitas dan isolasi kesalahan yang merupakan konsep penting dalam arsitektur microservices. Untuk memperkuat jawaban, Anda dapat menjelaskan bagaimana arsitektur microservices memungkinkan penggunaan teknologi yang berbeda-beda untuk setiap layanan dan bagaimana isolasi kesalahan dapat dilakukan dengan membatasi dampak kesalahan pada	Valid	Valid



© 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

		layanan tertentu saja.		
Apa yang dimaksud dengan pengujian regresi dalam konteks pengembangan perangkat lunak dan bagaimana tujuannya?	Pengujian rekreasi dalam konteks pengembangan perangkat lunak yaitu menguji bagaimana angka direkreasi. Dan tujuannya yaitu untuk mengetahui berapa hasil dari rekreasi kedepannya.	Jawaban Anda belum membahas konsep verifikasi dan stabilitas dalam pengujian regresi. Untuk memperkuat jawaban, Anda dapat menjelaskan proses verifikasi bahwa perubahan kode baru tidak mempengaruhi negatif fitur atau fungsionalitas yang sudah ada dalam sistem, sehingga memastikan stabilitas dan kinerja sistem setelah pembaruan, seperti memeriksa pengaruh perubahan kode pada fungsionalitas yang sudah ada.	Valid	Valid
Bagaimana cara mengukur efektivitas pengujian kode	Dalam pengembangan perangkat lunak, hal yang paling penting adalah memilih	Jawaban Anda belum membahas metode cakupan kode dan persentase kode yang telah	Valid	Valid



© 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

dengan menggunakan metode cakupan kode, dan apa saja manfaatnya dalam meningkatkan kualitas perangkat lunak?	bahasa pembagaman yang sesuai dengan kebutuhan proyek, misalnya menggunakan Python, Java, atau JavaScript. Selain itu, pengembang juga perlu memastikan bahwa aplikasi memiliki kata buka yang menarik, basis data yang stabil, serta dokumentasi yang lengkap, agar model dipelajari dalam anggota team lainnya. Penggunaan framework modernnya seperti Laravel, Spring Boot, atau React juga dapat membantu mempercepat proses pengembangan karena telah menyediakan berbagai fitur yang siap digunakan.	diuji. Untuk memperkuat jawaban, Anda dapat menjelaskan konsep cakupan kode dengan contoh seperti menggunakan tools seperti JaCoCo atau Cobertura untuk menganalisis cakupan kode dan menentukan area yang belum teruji.		
Apa perbedaan antara array	Array itu untuk menampilkan deretan angka. Kalau	Jawaban Anda belum membahas perbedaan	Valid	Valid



© 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 linked list dalam hal penggunaan memori dan akses elemen?	LinkedList itu untuk menampilkan deretan list.	penggunaan memori dan akses elemen antara array dan linked list. Untuk memperkuat jawaban, perlu menjelaskan konsep node, traversal, dan indeks, seperti menjelaskan bahwa array menggunakan indeks untuk akses elemen, sedangkan linked list menggunakan node yang saling terhubung untuk akses elemen.		
---	--	--	--	--

Tabel 26 merupakan hasil pengujian validitas feedback yang dihasilkan oleh sistem melalui metode human evaluation. Pengujian dilakukan terhadap 15 sampel yang berisi pertanyaan wawancara, jawaban kandidat, dan feedback AI. Seluruh sampel dinilai oleh dua orang pakar di bidang IT, yaitu seorang IT Senior dan seorang Manager. Pada beberapa sampel yang awalnya mendapat penilaian berbeda antar pakar, dilakukan diskusi lanjutan untuk mencapai kesepakatan penilaian akhir. Hasil evaluasi menunjukkan bahwa 10 dari 15 sampel dinilai valid, sedangkan 5 sampel dinilai belum valid. *Feedback* yang dikategorikan belum valid umumnya disebabkan oleh sistem yang keliru merepresentasikan isi jawaban kandidat dalam beberapa kasus, sistem menyatakan suatu konsep belum dibahas padahal konsep tersebut sudah dijelaskan kandidat, sehingga diperlukan penyempurnaan instruksi pada modul feedback agar sistem melakukan verifikasi silang terhadap isi jawaban kandidat sebelum menyimpulkan suatu konsep belum dibahas.



3. Black Box Testing

Pengujian ini berfokus pada spesifikasi fungsional dari perangkat lunak tanpa perlu mengetahui struktur internal atau kode sumber yang ada di baliknya. Penguji akan memberikan berbagai variasi input ke dalam sistem dan mengamati output yang dihasilkan untuk memastikan bahwa setiap fitur merespons dengan benar dan sesuai dengan hasil yang diharapkan. Berikut ini hasil pengujian *Black Box Testing* dapat dilihat pada Tabel 27.

Tabel 27 Hasil Pengujian Black Box Testing

Test Case ID	Test Case Name	Test Condition	Test Case Desc	Expected Result	Actual Result	Test Case Status
TC_POS_Login User_1	Verifikasi login dengan kredensial valid	Positif	Pengguna memasukkan Nama Pengguna dan Kata Sandi yang benar dan telah terdaftar	Sistem mengautentikasi pengguna dan mengarahkan ke halaman utama	Sistem mengautentikasi pengguna dan mengarahkan ke halaman utama	Sesuai
TC_NEG_Login User_2	Verifikasi login dengan kata sandi salah	Negatif	Pengguna memasukkan Nama Pengguna yang benar, tetapi memasukkan Kata Sandi yang salah	Sistem menolak akses dan memunculkan alert peringatan bahwa kata sandi salah	Sistem memunculkan alert peringatan bahwa kata sandi salah	Sesuai
TC_NEG_Login User_3	Verifikasi login dengan nama pengguna tidak terdaftar	Negatif	Pengguna memasukkan Nama Pengguna yang salah/tidak terdaftar di sistem	Sistem menolak akses dan memunculkan alert peringatan bahwa nama pengguna tidak	Sistem memunculkan alert peringatan bahwa nama pengguna tidak	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



© 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

				ditemukan/salah	ditemukan/salah	
TC_NEG_Login User_4	Verifikasi login dengan mengosongkan field	Negatif	Pengguna membiarkan kolom Nama Pengguna dan Kata Sandi kosong	Sistem menolak akses dan memunculkan alert peringatan bahwa kolom wajib diisi	Sistem memunculkan alert peringatan bahwa kolom wajib diisi	Sesuai
TC_POS_Register User_1	Verifikasi register dengan menggunakan data valid	Positif	Pengguna mengisi semua kolom (Nama Pengguna, Email, Kata Sandi, Konfirmasi) dengan format yang benar	Pendaftaran berhasil, akun terbentuk, dan sistem mengarahkan pengguna ke halaman login	Akun terbentuk dan pengguna masuk ke halaman login	Sesuai
TC_NEG_Register User_2	Verifikasi register dengan mengosongkan field wajib	Negatif	Pengguna membiarkan salah satu atau semua kolom.	Sistem menolak pendaftaran dan memunculkan alert peringatan bahwa kolom tidak boleh kosong.	Sistem memunculkan alert peringatan bahwa kolom tidak boleh kosong.	Sesuai
TC_NEG_Register User_3	Verifikasi register dengan format email tidak valid	Negatif	Pengguna mengisi Alamat Email tanpa menggunakan format standar	Sistem menolak pendaftaran dan memunculkan alert peringatan format email tidak valid	Sistem memunculkan alert peringatan format email tidak valid	Sesuai



© 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

TC_NEG_ Register User_4	Verifikasi register saat Kata Sandi dan Konfirmasi Kata Sandi berbeda	Negatif	Pengguna memasukkan Kata Sandi yang berbeda dengan input pada kolom Konfirmasi Kata Sandi	Sistem menolak pendaftaran dan memunculkan alert peringatan bahwa kata sandi tidak cocok	Sistem memunculkan alert peringatan bahwa kata sandi tidak cocok	Sesuai
TC_POS_ Room_1	Verifikasi memulai wawancara	Positif	Pengguna memilih Kategori, lalu memilih Peran, lalu memilih Tingkatan secara berurutan. Setelah semua terisi, pengguna menekan tombol "Mulai Wawancara"	Tombol "Mulai Wawancara" menjadi aktif, dan setelah ditekan, pengguna diarahkan ke sesi wawancara	Tombol aktif dan pengguna diarahkan ke sesi wawancara	Sesuai
TC_NEG_ Room_2	Verifikasi validasi tombol "Mulai Wawancara"	Negatif	Pengguna baru mengisi satu atau dua parameter atau tidak mengisi sama sekali	Tombol "Mulai Wawancara" tetap dalam keadaan tidak aktif (disabled / tidak dapat diklik) untuk mencegah masuk ke sesi tanpa parameter lengkap	Tombol "Mulai Wawancara" dalam keadaan tidak aktif	Sesuai
TC_NEG_ Room_3	Verifikasi urutan menu	Negatif	Pengguna langsung membuka menu	Menu tarik-turun Peran dan Tingkatan	Menu tarik-turun Peran dan	Sesuai



© 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

	tarik-turun (Dropdown)		tarik-turun Peran atau Tingkatan sebelum memilih Kategori	dalam keadaan kosong (tidak menampilkan daftar apa pun) karena bergantung pada pilihan Kategori	Tingkatan kosong/tidak menampilkan daftar	
TC_NEG_ Room_4	Verifikasi pemuatan data dinamis pada menu tarik-turun	Negatif	Pengguna memilih "Kategori", memastikan daftar "Peran" muncul. Lalu memilih "Peran", dan memastikan daftar "Tingkatan" muncul dengan benar	Data pada menu Peran dan Tingkatan termuat dan menyesuaikan secara otomatis berdasarkan pilihan sebelumnya	Data menu Peran dan Tingkatan muncul sesuai dengan Kategori yang dipilih	Sesuai
TC_POS_I nterview_1	Verifikasi pengambilan dan penampilan pertanyaan dari AI	Positif	Pengguna berhasil masuk ke ruang wawancara dan sistem menginisiasi sesi tanya jawab	Sistem berhasil mengambil data konteks dari VectorDB dan menampilkan teks pertanyaan	Pertanyaan dari AI berhasil ditampilkan	Sesuai
TC_POS_I nterview_2	Verifikasi pengiriman jawaban dan penyimpanan database	Positif	Pengguna memasukkan jawaban ke dalam kolom input dan menekan tombol kirim/lanjut	Jawaban pengguna ditampilkan pada antarmuka percakapan, data berhasil tersimpan ke dalam database	Jawaban tertampil, tersimpan, dan AI merespons dengan pertanyaan baru	Sesuai



© 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

				PostgreSQL, dan sistem memicu AI untuk memberikan pertanyaan selanjutnya		
TC_POS_I nterview_3	Verifikasi penghentian sesi otomatis saat batas waktu habis	Positif	Pengguna berada di dalam sesi wawancara dan membiarkan waktu hitung mundur (countdown timer) berjalan hingga mencapai 00:00	Sistem secara otomatis menghentikan sesi wawancara, menyimpan data terakhir, dan langsung mengarahkan pengguna ke halaman Hasil Wawancara	Sesi otomatis berhenti dan berpindah ke halaman Hasil Wawancara	Sesuai
TC_POS_I nterview_4	Verifikasi pengakhiran sesi secara manual	Positif	Pengguna menekan tombol "Akhir Wawancara" sebelum batas waktu hitung mundur habis	Sistem meminta konfirmasi, lalu menghentikan sesi, memproses data, dan mengarahkan pengguna ke halaman Hasil Wawancara	Sesi berhenti dan pengguna diarahkan ke halaman Hasil Wawancara	Sesuai
TC_POS_ Result_1	Verifikasi penampilan Analisis & Laporan Akhir	Positif	Pengguna membuka halaman hasil untuk melihat ringkasan performa secara keseluruhan	Sistem menampilkan komponen Summary, Kekuatan Teknis, Kelemahan,	Laporan analisis akhir tertampil lengkap dan terstruktur	Sesuai



© 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 Rekomendasi Belajar yang dihasilkan secara otomatis oleh AI		
TC_POS_ Result_2	Verifikasi tampilan Riwayat Transkrip Lengkap	Positif	Pengguna menggulir halaman ke bawah untuk meninjau kembali detail percakapan	Sistem menampilkan seluruh teks transkrip antara pewawancara dan kandidat	Seluruh transkrip percakapan tertampil dengan rapi	Sesuai
TC_POS_ Result_3	Verifikasi skor dan evaluasi spesifik per jawaban	Positif	Pengguna meninjau kotak "Technical Gap & Evaluasi AI" pada setiap butir jawaban di transkrip	Sistem menampilkan skor numerik dan feedback tekstual yang spesifik sesuai dengan jawaban yang diberikan kandidat pada poin tersebut	Skor dan feedback per jawaban muncul sesuai hasil olahan AI	Sesuai
TC_POS_ Result_4	Verifikasi integrasi data dari Riwayat	Positif	Pengguna memilih salah satu data lama dari menu "Interview History" untuk melihat kembali hasilnya	Sistem berhasil memanggil data dari PostgreSQL dan menampilkan kembali hasil wawancara yang sama persis dengan saat sesi baru berakhir	Data lama berhasil dipanggil dan ditampilkan kembali dengan benar	Sesuai

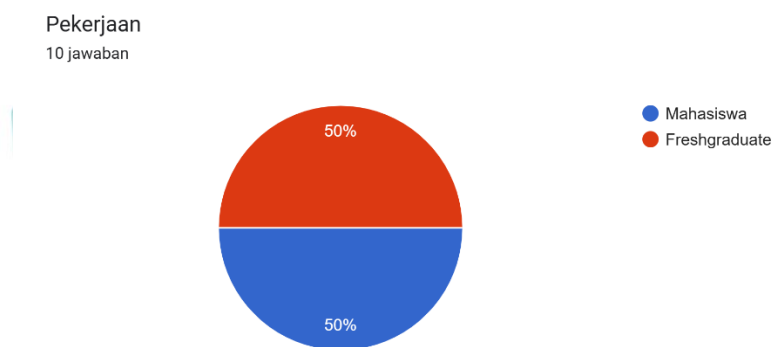


Hak Cipta :

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

4. System Usability Scale

Pengujian ini melibatkan 10 orang responden yang diminta untuk mengisi kuesioner setelah mereka menguji coba langsung aplikasi simulasi wawancara. Namun, sebelum masuk ke pembahasan analisis skor SUS, perlu dijabarkan terlebih dahulu profil demografi dari para responden yang terlibat. Pemetaan demografi ini dikelompokkan berdasarkan dua aspek utama, yaitu rentang usia dan latar belakang pekerjaan. Rincian mengenai persentase status pekerjaan para responden tersebut dapat dilihat pada Gambar 4.38 berikut.



Gambar 4.38 Responden Status Pekerjaan

Gambar 4.38 menunjukkan demografi responden berdasarkan latar belakang pekerjaan. Merujuk pada diagram tersebut, dapat diketahui bahwa proporsi responden terbagi secara merata, yaitu sebanyak 50% (5 orang) berstatus sebagai Mahasiswa dan 50% (5 orang) lainnya merupakan Fresh Graduate (lulusan baru). Komposisi ini sangat relevan dengan tujuan evaluasi, mengingat kedua kelompok ini adalah target pengguna utama yang akan paling banyak memanfaatkan aplikasi InterviewMate sebagai sarana latihan untuk menghadapi wawancara kerja yang sesungguhnya.

Setelah profil demografi responden dipastikan sesuai dengan target sasaran aplikasi, tahap selanjutnya adalah mengolah data kuesioner *System Usability Scale* (SUS) yang telah dikumpulkan. Untuk memudahkan penyajian dan pembacaan data, rincian hasil pengisian kuesioner dari kesepuluh responden tersebut disajikan



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

secara terpisah. Hasil pengisian kuesioner dapat dilihat pada Tabel 28 untuk hasil pengujian SUS.

Tabel 28 Hasil Pengujian SUS

Responden	Jawaban Kuesioner SUS									
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
Responden 1	5	1	5	1	5	1	5	1	5	1
Responden 2	5	2	5	1	5	1	5	1	1	1
Responden 3	4	1	5	1	5	1	5	1	5	1
Responden 4	5	1	5	1	5	1	5	2	5	1
Responden 5	4	2	4	1	4	1	4	1	4	1
Responden 6	5	1	5	1	4	2	5	1	5	1
Responden 7	4	2	4	3	4	2	4	2	4	2
Responden 8	4	2	4	1	4	4	3	1	5	1
Responden 9	5	3	5	2	5	1	5	2	5	3
Responden 10	5	1	5	1	5	5	1	5	1	5

Setelah data responden diperoleh, dilakukan transformasi skor sesuai aturan *System Usability Scale* (SUS), yaitu untuk pernyataan bernomor ganjil skor dihitung dengan mengurangi nilai responden dengan 1, sedangkan untuk pernyataan bernomor genap skor dihitung dengan mengurangi nilai responden dari 5. Selanjutnya, seluruh skor dijumlahkan dan dikalikan 2,5 untuk memperoleh nilai SUS pada skala 0–100. Rata-rata skor seluruh responden kemudian dihitung untuk menentukan tingkat kebergunaan aplikasi InterviewMate. Hasil perhitungan tersebut disajikan pada Tabel 29.

Tabel 29 Hasil Perhitungan SUS

Skor Hasil Hitung										Jumlah	Nilai (Jumlah x 2.5)
Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
4	4	4	4	4	4	4	4	4	4	40	100
4	3	4	4	4	4	4	4	0	4	35	87,5
3	4	4	4	4	4	4	4	4	4	39	97,5
4	4	4	4	4	4	4	3	4	4	39	97,5
3	3	3	4	3	4	3	4	3	4	34	85
4	4	4	4	3	3	4	4	4	4	38	95
3	3	3	2	3	3	3	3	3	3	29	72,5
3	3	3	4	3	1	2	4	4	4	31	77,5
4	2	4	3	4	4	4	3	4	2	34	85
4	4	4	4	4	4	4	4	4	4	40	100
Skor Rata-rata (Hasil Akhir)											89,75



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

Berdasarkan hasil pengolahan data pada Tabel 29, aplikasi InterviewMate memperoleh skor rata-rata *System Usability Scale* (SUS) sebesar 89,75. Berdasarkan interpretasi SUS, skor tersebut termasuk dalam kategori *Grade A* dengan *Adjective Rating Excellent*, yang menunjukkan bahwa aplikasi memiliki tingkat kebergunaan yang sangat baik dan dapat diterima oleh pengguna.

5. Net Promoter Score (NPS)

Pengujian *Net Promoter Score* (NPS) dilakukan untuk mengukur tingkat loyalitas serta kesediaan pengguna dalam merekomendasikan aplikasi InterviewMate kepada orang lain. Pengujian melibatkan 10 responden yang memberikan penilaian pada skala 1–10 setelah mencoba seluruh fitur simulasi wawancara. Ringkasan hasil pengujian NPS disajikan pada Tabel 30.

Tabel 30 Hasil Pengujian NPS

Skala	0	1	2	3	4	5	6	7	8	9	10
Jumlah Jawaban	0	0	0	0	0	0	0	0	1	6	3
Kategori	Detractor							Passive		Promoter	
Persentase	0%							10%		90%	
NPS	90%										

Berdasarkan hasil pengujian NPS pada Tabel 30, dari 10 responden tidak terdapat pengguna yang termasuk kategori *Detractor* (0%), sebanyak 1 responden (10%) termasuk kategori *Passive*, dan 9 responden (90%) termasuk kategori *Promoter*. Berdasarkan perhitungan tersebut, diperoleh skor NPS sebesar 90, yang termasuk dalam kategori *World Class*. Hasil ini menunjukkan bahwa aplikasi InterviewMate mampu memberikan pengalaman penggunaan yang sangat memuaskan, sehingga mayoritas pengguna bersedia merekomendasikannya kepada orang lain sebagai media latihan wawancara kerja.

4.4.4 Analisis Data Pengujian

Tahap ini membahas analisis terhadap seluruh hasil pengujian sistem yang meliputi evaluasi akurasi kecerdasan buatan menggunakan RAGAs, validitas feedback AI melalui Human Evaluation, pengujian fungsionalitas menggunakan Black Box Testing, serta tingkat kepuasan dan loyalitas pengguna berdasarkan metode *System Usability Scale* (SUS) dan *Net Promoter Score* (NPS).



© 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

1. Analisis Pengujian RAGAs

Berdasarkan hasil evaluasi menggunakan framework RAGAs, arsitektur RAG pada sistem InterviewMate menunjukkan kinerja yang sangat baik dengan seluruh metrik memperoleh nilai di atas 0,90. Nilai *Answer Relevance* sebesar 0,9854 menunjukkan bahwa sistem mampu menghasilkan pertanyaan lanjutan yang relevan dengan konteks wawancara. Nilai *Faithfulness* sebesar 0,9413 membuktikan bahwa respons yang dihasilkan tetap mengacu pada basis pengetahuan sehingga meminimalkan terjadinya halusinasi. Sementara itu, nilai *Context Precision* sebesar 0,9200 menunjukkan bahwa mekanisme *retrieval* mampu mengambil informasi yang paling relevan dari *knowledge base*. Hasil ini menunjukkan bahwa proses *retrieval* dan kemampuan generatif LLM telah terintegrasi dengan baik sehingga sistem mampu menghasilkan simulasi wawancara yang kontekstual, relevan, dan *faktual*.

2. Analisis Pengujian Human Evaluation

Berdasarkan hasil Human Evaluation, sebanyak 10 dari 15 sampel dinilai valid oleh kedua pakar, sedangkan 5 sampel dinilai belum valid. Hasil ini menunjukkan bahwa sebagian besar umpan balik yang dihasilkan sistem telah sesuai dengan jawaban kandidat. Sementara itu, umpan balik yang belum valid umumnya disebabkan oleh sistem yang belum mampu mengidentifikasi informasi yang telah disampaikan kandidat secara tepat, sehingga menghasilkan saran yang kurang sesuai dengan konteks jawaban.

3. Analisis Pengujian Black Box Testing

Berdasarkan hasil pengujian *Black Box Testing*, seluruh skenario *test case* yang telah dirancang berhasil dijalankan sesuai dengan hasil yang diharapkan. Pengujian dilakukan terhadap 20 skenario yang terdiri atas pengujian positif dan pengujian negatif, dengan tingkat keberhasilan mencapai 100%. Hasil ini menunjukkan bahwa seluruh fungsionalitas sistem telah berjalan dengan baik sesuai kebutuhan.

4. Analisis Pengujian System Usability Scale (SUS)

Berdasarkan hasil perhitungan *System Usability Scale* (SUS), aplikasi InterviewMate memperoleh skor rata-rata sebesar 89,75. Berdasarkan pedoman interpretasi SUS, skor tersebut termasuk dalam kategori *Acceptable*, dengan Grade A dan *Adjective Rating Excellent*. Hasil ini menunjukkan bahwa aplikasi memiliki



© 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

tingkat kebergunaan yang sangat baik, mudah dipelajari, serta mampu memberikan pengalaman penggunaan yang nyaman bagi pengguna.

5. Analisis Pengujian Net Promoter Score (NPS)

Berdasarkan hasil perhitungan *Net Promoter Score* (NPS), aplikasi InterviewMate memperoleh skor sebesar 90, yang termasuk dalam kategori sangat baik. Hasil ini menunjukkan bahwa aplikasi mampu memberikan pengalaman penggunaan yang memuaskan serta memenuhi kebutuhan pengguna dalam berlatih wawancara kerja. Tingginya persentase responden yang termasuk dalam kategori *Promoter* menunjukkan bahwa sebagian besar pengguna bersedia merekomendasikan aplikasi InterviewMate kepada orang lain sebagai media latihan wawancara kerja.



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

Penelitian ini telah berhasil mengimplementasikan sistem wawancara kerja otomatis dengan metode *Retrieval-Augmented Generation* (RAG). Dalam sistem ini, *Large Language Model* (LLM) berfungsi sebagai mesin penggerak utama dalam pembuatan pertanyaan yang dipadukan dengan sistem RAG untuk mengambil data kontekstual dari ChromaDB. Keberhasilan implementasi sistem ini secara langsung telah menjawab dan merealisasikan manfaat-manfaat yang ditargetkan pada penelitian, yaitu:

1. Fitur Otomatisasi Pertanyaan, Berfungsi untuk memberikan gambaran yang akurat mengenai potensi pertanyaan teknis sesuai dengan posisi yang dilamar, yang secara langsung memudahkan pelamar dalam melatih kesiapan wawancara kerja secara mandiri dan terarah.
2. Fitur Interaksi Berbasis Suara, Berfungsi untuk melatih mental dan membangun rasa percaya diri bagi *interviewee*, karena sistem mampu menghadirkan pengalaman simulasi wawancara yang interaktif dan realistis layaknya berbicara langsung dengan pewawancara manusia.
3. Fitur Analisis dan Laporan Evaluasi Otomatis, Berfungsi untuk memberikan umpan balik yang komprehensif dan objektif bagi *interviewee*. Melalui fitur ini, pengguna dapat mengetahui letak kekurangan dari jawabannya untuk diperbaiki di kemudian hari.

Melalui sistem ini, LLM memegang peran penting untuk menjaga agar setiap pertanyaan yang keluar selalu relevan. Selain itu, proses penilaiannya juga menjadi lebih cerdas karena jalurnya diarahkan langsung oleh konteks dari RAG.

Penelitian ini juga telah mendapatkan hasil evaluasi menggunakan RAGAs untuk melihat performa sistem yang diberikan pada metrik-metrik utama seperti relevansi jawaban (*answer relevance*) dan ketepatan konteks (*context precision*). Evaluasi ini membuktikan bahwa sistem mampu mengambil informasi secara akurat dan menghasilkan luaran yang sesuai tanpa menyimpang dari sumber data.



© 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

Selanjutnya, dari segi fungsionalitas, hasil pengujian *Black Box* menunjukkan bahwa seluruh fitur pada aplikasi InterviewMate telah berjalan dengan baik dan sesuai dengan rancangan awal yang diharapkan tanpa adanya kegagalan sistem. Terakhir, dari sisi penerimaan dan kepuasan pengguna, pengujian menggunakan *System Usability Scale* (SUS) memperoleh skor rata-rata akhir sebesar 91 yang masuk ke dalam kategori Grade A dengan predikat *Excellent*.

Melengkapi hasil tersebut, pengujian tingkat loyalitas menggunakan *Net Promoter Score* (NPS) juga berhasil meraih skor akhir sebesar 90. Gabungan dari keseluruhan evaluasi ini menegaskan bahwa aplikasi InterviewMate tidak hanya canggih secara teknis, fungsional, dan sangat mudah digunakan, tetapi juga sukses membangun kepuasan tingkat tinggi, di mana para pencari kerja secara sukarela dan tanpa ragu bersedia merekomendasikan aplikasi ini kepada rekan mereka untuk berlatih wawancara.

5.2 Saran

Berdasarkan hasil penelitian, pengujian, serta evaluasi yang telah dilakukan terhadap aplikasi InterviewMate, terdapat beberapa keterbatasan sistem yang dapat dijadikan acuan untuk pengembangan pada penelitian selanjutnya, antara lain:

1. Integrasi Analisis Video dan Ekspresi Wajah, Pengembangan selanjutnya disarankan untuk mengintegrasikan analisis visual. Dengan memanfaatkan *Computer Vision (Facial Emotion Recognition)*, sistem tidak hanya menilai suara, tetapi juga mampu mendeteksi tingkat kegugupan, fokus mata, dan bahasa tubuh pengguna. Penambahan ini akan membuat laporan evaluasi jauh lebih komprehensif, mengingat komunikasi non-verbal sangat krusial dalam wawancara kerja yang sebenarnya.
2. Peningkatan Kualitas Suara AI (*Text-to-Speech*), Untuk menciptakan atmosfer simulasi yang terasa lebih nyata, pengembangan ke depan disarankan untuk meningkatkan kualitas audio dari pewawancara AI. Penggunaan layanan *Text-to-Speech* (TTS) premium yang berbayar seperti ElevenLabs atau OpenAI TTS API bisa menjadi opsi terbaik. Melalui API premium tersebut, intonasi, jeda napas, dan artikulasi suara AI akan terdengar jauh lebih alami dan emosional layaknya manusia asli, sehingga bisa menghilangkan kesan kaku atau robotik saat pengguna sedang diwawancarai.



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

DAFTAR PUSTAKA

- Albert, G. D., & Voutama, A. (2025). *PDF MENGGUNAKAN LOCAL RETRIEVAL-AUGMENTED GENERATION (RAG) DAN OLLAMA*. 13(2).
- Angelina, Yandhika, C., Hartanto, C. L., Graciela, M., & Farisi, A. (2024). *Sebuah Tinjauan Literatur Sistematis Tentang Metode Pengembangan Perangkat Lunak Sistem Informasi Berbasis Web*. 5(1), 181–192.
- Anwar, M. R., Mashudi, I. A., & Yunhasnawa, Y. (2025). *A Study on the Performance of a RAG- Augmented Interview Chatbot*.
- Babu, S., & Krishnan, P. (2025). *International Journal of Innovative Research in Science Engineering and Technology (IJIRSET) LLM Evaluations : A Survey of Programmatic , Human , and LLM-as-Judge Approaches*. 14(6), 6–11. <https://doi.org/10.15680/IJIRSET.2025.1406011>
- Balaguer, A., Benara, V., Cunha, R., Estevão, R., Hendry, T., Holstein, D., Marsman, J., Mecklenburg, N., Malvar, S., Nunes, L. O., Padilha, R., Sharp, M., Silva, B., Sharma, S., Aski, V., & Chandra, R. (2023). *VS Microsoft*.
- Binangkit, C. A., Voutama, A., Heryana, N., Komputer, F. I., Karawang, U. S., & Musik, A. (2023). *PEMANFAATAN UML (UNIFIED MODELING LANGUAGE) DALAM PERENCANAAN SISTEM PENGELOLAAN SEWA ALAT MUSIK BERBASIS*. 7(2), 1429–1436.
- Brown, C., Tech, V., & Tech, V. (2025). *Towards Evidence-Based Tech Hiring Pipelines*. In *Proceedings of International Workshop on Software Engineering in 2030 (SE 2030)* (Vol. 1, Issue 1). Association for Computing Machinery.
- Budiasningrum, R. S., Rosita, R., Setiawan, J., Yuliana, D., & Astuti, E. D. (2021). *TEKNIK JITU MEMENANGKAN TAHAPAN TES WAWANCARA*. 1, 49–53.
- Giffar, Z. M., Adnan, I. Z., Hendrawan, H., Studi, P., Komunikasi, I., Garut, U., Garut, K., Schutz, A., Komunikasi, K., Kerja, W., Schutz, A., Anxiety, C., & Interview, J. (2024). *MAKNA KECEMASAN KOMUNIKASI DALAM WAWANCARA KERJA BAGI*. 9(4), 810–828.
- Guo, S., Liao, L., Li, C., & Chua, T. (2021). *A Survey on Neural Question Generation: Methods, Applications, and Prospects*.

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

- Hu, R., Liu, S., Qi, P., Liu, J., & Li, F. (2025). ICCA-RAG : Intelligent Customs Clearance Assistant Using Retrieval-Augmented Generation (RAG). *IEEE Access*, 13(January), 39711–39726. <https://doi.org/10.1109/ACCESS.2025.3544408>
- Innayah, D., Pasha, S. R., Alifia, N. N., Ekonomi, F., & Pamulang, U. (2021). *Dampak Teknologi dan Inovasi dalam Manajemen Sumber Daya Manusia di Era Digital*. 439–445.
- Junaidi, S., Devegi, M., & Kurniawan, H. (2023). *Pelatihan Pengolahan dan Visualisasi Data Penduduk Menggunakan Python*. 4(1), 151–162. <https://doi.org/10.30812/adma.v4i1.2963>
- Kharisma, I. L., Hidayat, M. S., Somantri, & Kamdan. (2025). *Implementasi Retrieval Augmented Generation dalam Sistem Chatbot Dermatologi Berbasis Website Implementation of Retrieval Augmented Generation in a Web-Based Dermatological Chatbot System*. 11, 448–462.
- Kholifah, S. N., Heryana, N., & Nugraha, H. B. (2023). *ANALISIS USABILITY PADA APLIKASI HIMFO MENGGUNAKAN METODE SYSTEM USABILITY SCALE (SUS) (STUDI KASUS HIMPUNAN MAHASISWA TEKNIK INFORMATIKA UNSIKA)*. 7(2), 1416–1422.
- Lata, S. (2025). *Retrieval-Augmented Generation (RAG) and Memory Systems for HR and Enterprise AI*. 0–3.
- Li, M., Chen, X., Liao, W., Song, Y., Zhang, T. A. O., & Zhao, D. (2025). *EZInterviewer : To Improve Job Interview Performance with Mock EZInterviewer : To Improve Job Interview Performance with Mock Interview Generator*. February 2023. <https://doi.org/10.1145/3539597.3570476>
- Musthofa, K. N., & Haryono, W. (2023). *CUTI KARYAWAN BERBASIS WEB MENGGUNAKAN METODE SYSTEM DEVELOPMENT LIFE CYCLE (SDLC) PADA SD BUDI MULIA*. 1(3), 951–958.
- Pathak, Gangesh., Pandey, D. (2025). *AI Agents in Recruitment: A Multi-Agent System for Interview, Evaluation, and Candidate Scoring*.
- Praniffa, A. C., Syahri, A., Sandes, F., Fariha, U., & Giansyah, Q. A. (2023). *Jurnal Testing dan Implementasi Sistem Informasi PARKIR BERBASIS WEB BLACK BOX AND WHITE BOX TESTING OF WEB-BASED PARKING*. 1(1), 1–16.

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

- Pratama, R. P., & Avianto, D. (2025). *BULLETIN OF COMPUTER SCIENCE RESEARCH Implementasi Sistem Klasifikasi Batik Menggunakan MobileNet dengan Integrasi Chatbot Retrieval Augmented Generation*. 6(1), 75–84. <https://doi.org/10.47065/bulletincsr.v6i1.863>
- Rianti, A., Wachid, N., Majid, A., & Fauzi, A. (2023). *CRISP-DM : Metodologi Proyek Data Science*. 107–114.
- Rohmanto, R., & Setiawan, T. (2022). *Perbandingan Efektivitas Sistem Pembelajaran Luring dan Daring Menggunakan Metode Use case dan Sequence Diagram*. 5(1), 53–62.
- S, Y. A. P. G., Ginting, G. L., & Panjaitan, M. (2023). *Optimasi Penerimaan Siswa Baru dengan Penerapan Algoritma Text Mining dan TF-IDF*. 2(3), 109–115. <https://doi.org/10.47065/comforch.v2i3.941>
- Saha, B., Saha, U., & Member, G. S. (2024). QuIM-RAG : Advancing Retrieval-Augmented Generation With Inverted Question Matching for Enhanced QA Performance. *IEEE Access*, 12(December), 185401–185410. <https://doi.org/10.1109/ACCESS.2024.3513155>
- Siswipraptini, P. C., Wafit, M. I., & Djunaidi, K. (2023). *Klasifikasi Pekerjaan Bidang Teknologi Informasi Menggunakan Algoritma Cosine Similarity*. 12(1), 38–48.
- Sun, H., & Lin, H. (2025). MockLLM : A Multi-Agent Behavior Collaboration Framework for Online Job Seeking and Recruiting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3–7, 2025, Toronto, ON, Canada (Vol. 1, Issue 1)*. arXiv. <https://doi.org/10.1145/3711896.3737051>
- Syah, M. P., Wardani, A. P., & Idhom, M. (2025). *Perbandingan Representasi Teks Tf-Idf Dan Bert Terhadap Akurasi*. 5(1), 47–59.
- Tarigan, N. L. L., Surya, P., & Wijaya, M. (2024). *Jurnal Mantra : Manajemen Strategis Analisis Tingkat Loyalitas Konsumen Generasi Z terhadap Marketplace di Indonesia Menggunakan Metode NPS (Net Promoter Score)*. 1(01), 21–34.
- Tohir, H., Merlina, N., & Haris, M. (2024). *UTILIZING RETRIEVAL-AUGMENTED GENERATION IN LARGE LANGUAGE MODELS TO*



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

ENHANCE INDONESIAN LANGUAGE NLP. 10(2), 352–360.
<https://doi.org/10.33480/jitk.v10i2.5916>.INTRODUCTION

Wahyuni, F. (2023). *PERANCANGAN SISTEM INFORMASI KAS BERBASIS WEB DENGAN MENGGUNAKAN METODE WATERFALL* Fitri Wahyuni. 7(1), 138–143.

Wijoyo, H. (2022). *Analisis teknik wawancara (pengertian wawancara, bentuk-bentuk pertanyaan wawancara) dalam penelitian kualitatif bagi mahasiswa teologi dengan tema pekabaran injil melalui penerjemahan alkitab.* 1–10.

Yang, F., & Yang, M. (2023). *LongRoPE: Extending LLM Context Window Beyond 2 Million Tokens.*

Yazdani, N., Mahajan, A., & Ansari, A. (2025). *Zara : An LLM-based Candidate Interview Feedback System.* *Mujtaba* 2024, 1–11.

Zen, M., Rizal, C., & Eka, M. (2022). *Perancangan Sistem Informasi Desa Tomuan Holbung Menggunakan Metode Waterfall.* 9(2), 274–280.
<https://doi.org/10.30865/jurikom.v9i2.3986>

**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 RIWAYAT HIDUP PENULIS

Ahmad Zainul Mufid



Lahir di Jakarta, 12 April 2003. Lulus dari SDN Sukamaju 01 pada 2016, SMPN 12 Depok pada tahun 2019, SMK Taruna Bhakti pada tahun 2022. Saat ini menempuh Program Studi Diploma Empat Teknik Informatika di Politeknik Negeri Jakarta. Memiliki keahlian pada bidang Web Development, dan ketertarikan pada bidang Mobile Development, Backend, Machine Learning, serta Data Science.



**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

Lampiran 1 Wawancara



1. Bagaimana alur wawancara kerja yang biasanya diterapkan di perusahaan?
Jawaban: Menurut narasumber, proses wawancara kerja umumnya diawali dengan peninjauan biodata dan CV kandidat untuk memahami latar belakang pendidikan, pengalaman kerja, serta keterampilan yang dimiliki. Setelah itu dilakukan wawancara non-teknis untuk menilai kemampuan komunikasi, motivasi kerja, pengalaman organisasi, dan kesesuaian kandidat dengan budaya perusahaan. Selanjutnya dilakukan wawancara teknis yang berfokus pada kompetensi sesuai posisi yang dilamar. Pada beberapa posisi, kandidat juga diberikan studi kasus atau permasalahan tertentu untuk mengukur kemampuan analisis dan pemecahan masalah. Tahap akhir biasanya membahas minat kandidat terhadap posisi yang dilamar serta ekspektasi kerja.
2. Aspek apa saja yang paling diperhatikan saat menilai kandidat?
Jawaban: Aspek yang paling diperhatikan meliputi kemampuan komunikasi, pengalaman kerja atau proyek yang pernah dikerjakan, keterampilan teknis yang relevan dengan posisi, kemampuan pemecahan masalah, serta pengalaman organisasi atau aktivitas lain yang menunjukkan kemampuan bekerja sama dan beradaptasi. Selain itu, kesesuaian kandidat dengan kebutuhan perusahaan dan posisi yang dilamar juga menjadi pertimbangan penting dalam proses penilaian.



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. Menurut Anda, apakah pertanyaan wawancara dapat dibuat secara otomatis berdasarkan CV dan posisi yang dilamar? Jika ya, aspek apa yang paling penting untuk dijadikan dasar?

Jawaban: Menurut narasumber, pertanyaan wawancara dapat dibuat secara otomatis berdasarkan CV dan posisi yang dilamar. Dasar utama yang perlu digunakan adalah pengalaman kerja, keterampilan yang tercantum dalam CV, latar belakang pendidikan, serta kompetensi yang dibutuhkan oleh posisi tersebut. Selain itu, sistem perlu memperhatikan kata kunci tertentu dan konteks jawaban agar pertanyaan yang dihasilkan tetap relevan dengan profil kandidat dan kebutuhan perusahaan. Pertanyaan juga dapat dikategorikan menjadi aspek teknis, non-teknis, psikologis, maupun studi kasus sesuai kebutuhan evaluasi.

4. Dalam proses wawancara, sumber informasi apa saja yang biasanya Anda gunakan sebagai acuan (misalnya CV, job description, standar kompetensi, atau pengalaman sebelumnya)?

Jawaban: Narasumber menjelaskan bahwa sumber informasi utama yang digunakan adalah CV kandidat dan deskripsi pekerjaan (job description). Selain itu, standar kompetensi yang dibutuhkan oleh posisi tertentu juga menjadi acuan dalam menyusun pertanyaan dan melakukan penilaian. Informasi mengenai perusahaan, kebutuhan divisi, serta pengalaman sebelumnya dari kandidat juga digunakan untuk menggali kompetensi secara lebih mendalam dan kontekstual.

5. Seberapa penting penyesuaian pertanyaan berdasarkan jawaban kandidat sebelumnya dalam menilai kompetensi mereka?

Jawaban: Penyesuaian pertanyaan berdasarkan jawaban sebelumnya dinilai sangat penting karena memungkinkan pewawancara mengeksplorasi kompetensi kandidat secara lebih mendalam. Dengan pendekatan ini, pertanyaan dapat berkembang dari tingkat dasar ke tingkat yang lebih kompleks sesuai kemampuan kandidat. Metode tersebut membantu mengidentifikasi tingkat pemahaman, pengalaman nyata, serta kemampuan pemecahan masalah yang dimiliki kandidat secara lebih akurat dibandingkan penggunaan daftar pertanyaan statis.

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

6. Jika sistem AI memberikan ringkasan atau analisis hasil wawancara, informasi apa yang menurut Anda wajib ditampilkan agar tetap relevan bagi recruiter?

Jawaban: Menurut narasumber, sistem AI sebaiknya menampilkan skor atau hasil evaluasi berdasarkan kategori kompetensi yang diukur, seperti kemampuan teknis, kemampuan non-teknis, dan keterampilan pemecahan masalah. Selain itu, sistem perlu memberikan umpan balik yang menjelaskan kelebihan dan kekurangan kandidat, area yang perlu ditingkatkan, serta rekomendasi posisi pekerjaan yang sesuai dengan kompetensi yang dimiliki. Penyajian hasil dalam bentuk ringkasan yang mudah dipahami akan membantu recruiter dalam proses pengambilan keputusan.

Manager – Fixed Data Management and Analysis



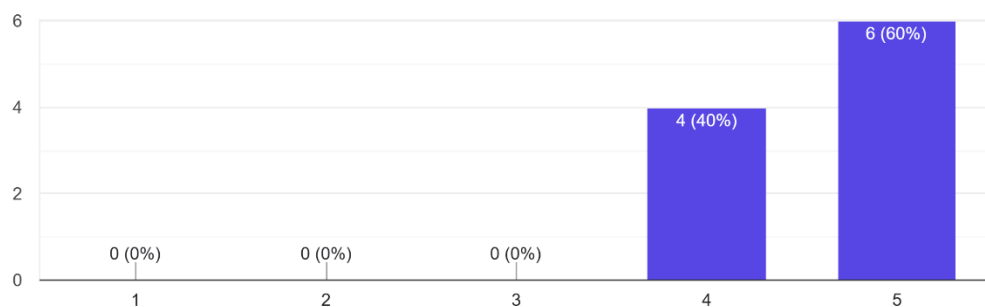
Teguh Arie Wibowo, S.T., M.M.

NIK. 85013

Lampiran 2 Hasil Kuesioner SUS

Saya berpikir akan menggunakan sistem ini lagi

10 jawaban

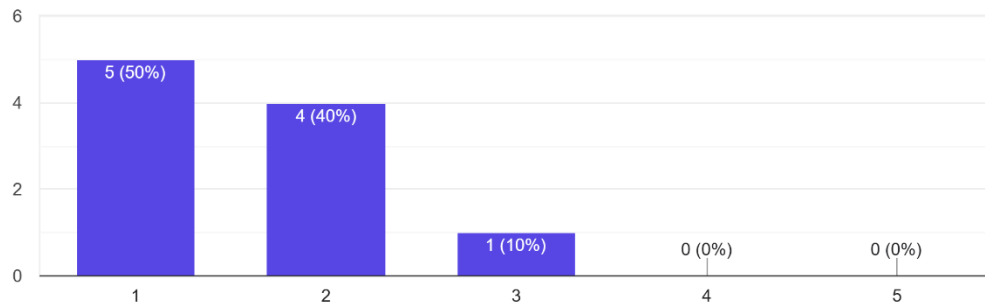


Hak Cipta :

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

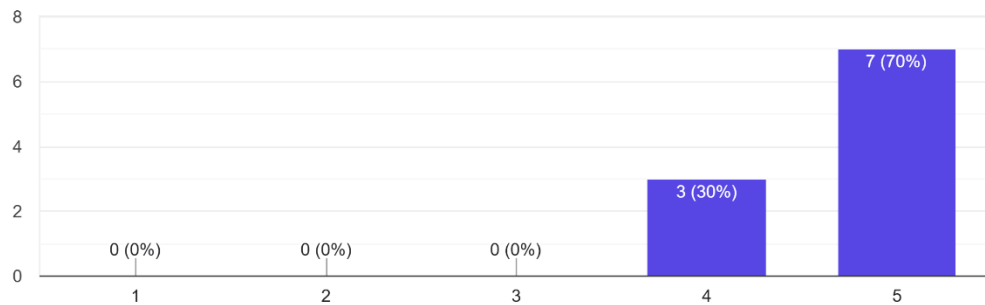
Saya merasa sistem ini rumit untuk digunakan

10 jawaban



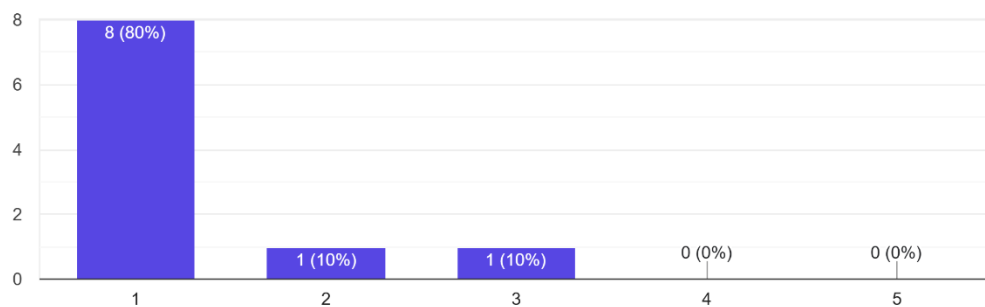
Saya merasa sistem ini mudah digunakan

10 jawaban



Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini

10 jawaban

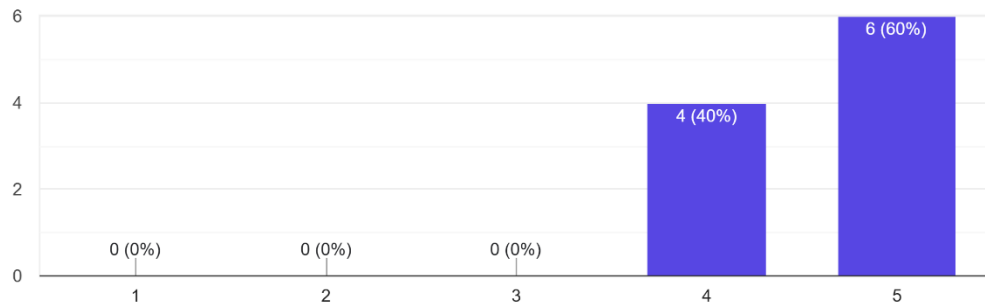


Hak Cipta :

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

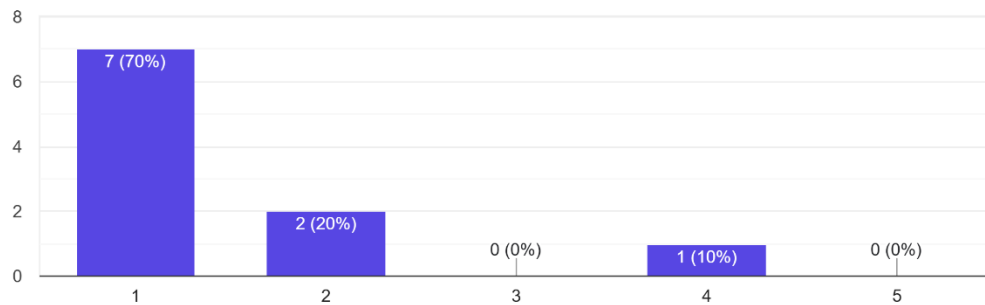
Saya merasa fitur-fitur sistem ini berjalan dengan semestinya

10 jawaban



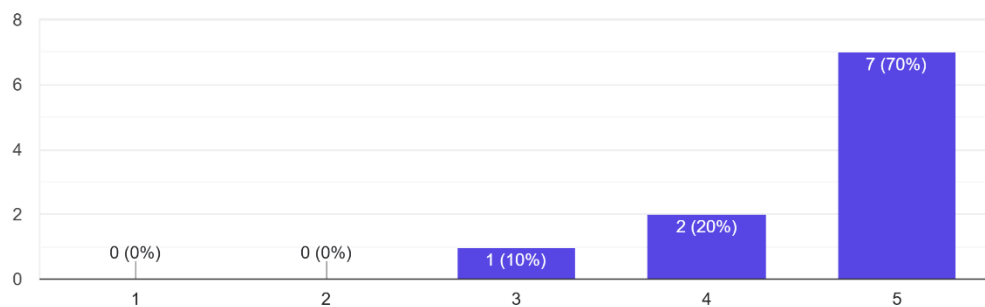
Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini)

10 jawaban



Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat

10 jawaban

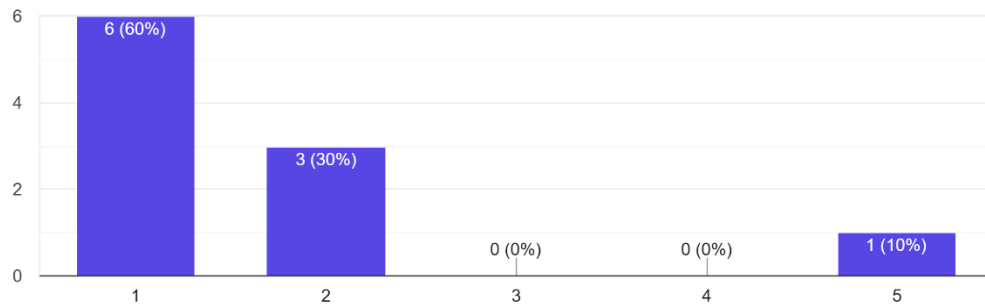


Hak Cipta :

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

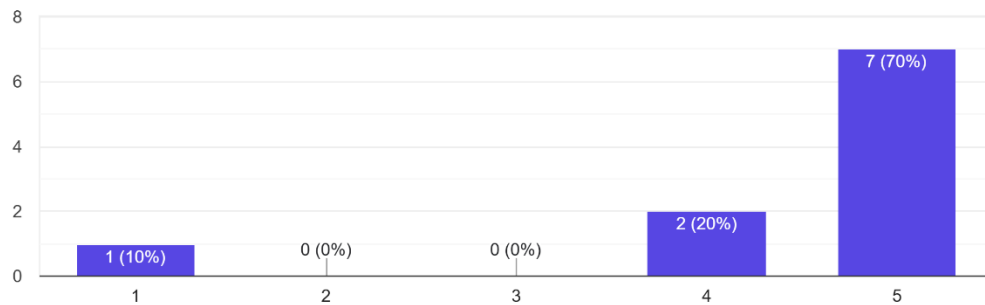
Saya merasa sistem ini membingungkan

10 jawaban



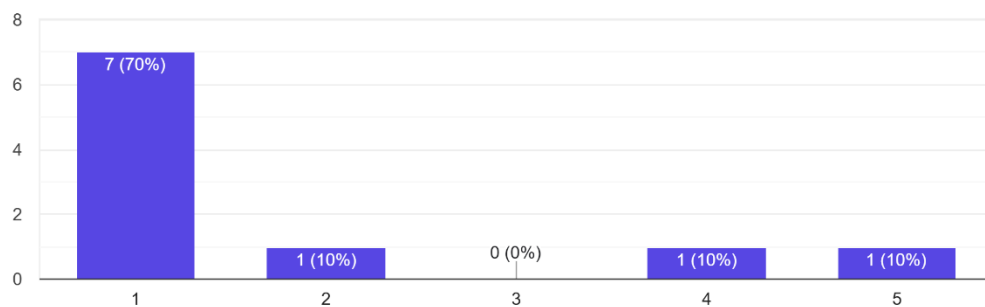
Saya merasa tidak ada hambatan dalam menggunakan sistem ini

10 jawaban



Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini

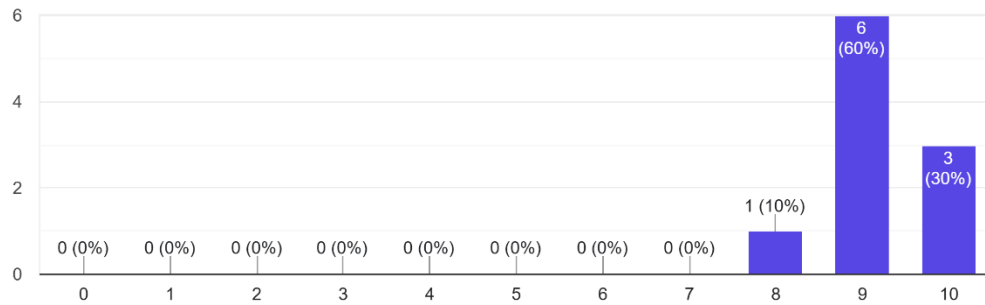
10 jawaban



Lampiran 3 Hasil Kuesioner NPS

Seberapa besar kemungkinan Anda akan merekomendasikan aplikasi InterviewMate kepada teman, kerabat Anda?

10 jawaban



Lampiran 4 Pengujian Aplikasi

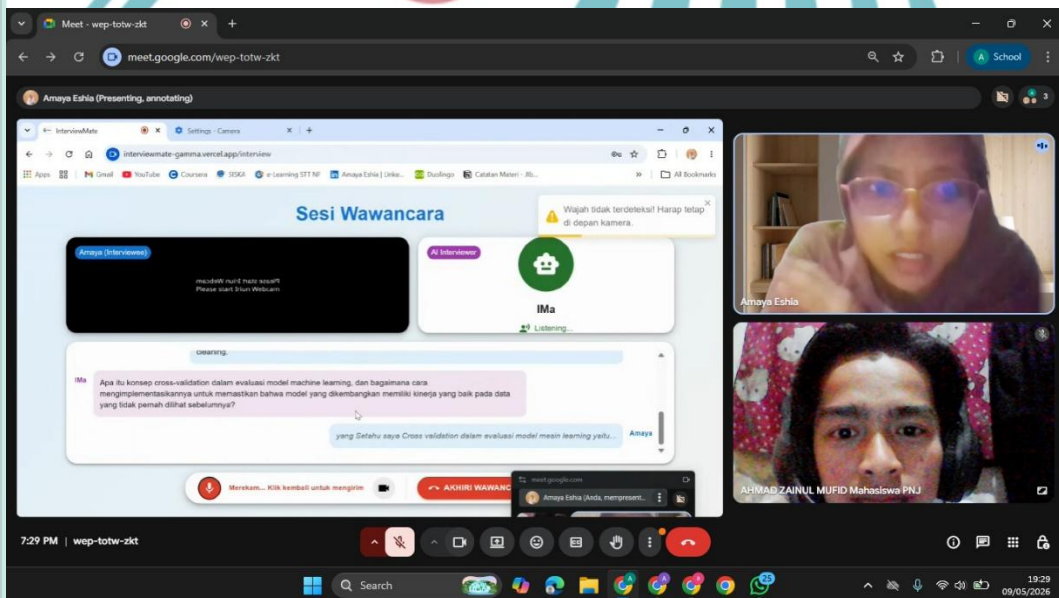




© Hak Cipta milik Politeknik Negeri Jakarta

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





© 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

