Journal of Innovative and Creativity
Journal Homepage: https://joecy.org/index.php/joecy
e-mail: journaljoecy@gmail.com

Faculty of Education
University of Pahlawan Tuanku Tambusai
ISSN 2775-771X (Online)
ISSN 2962-570X (Printed)
5 (2) 2025, Page: 20515-20523

Pengembangan Chatbot Multimodal Teks–Suara untuk Pelayanan Nasabah Bank Nagari

Khalilul Afwan[1], Resmi Darni[2],
Ahmaddul Hadi[3], Agariadne Dwinggo Samala[4]
Program Studi S1 Informatika Universitas Negeri Padang

Corresponding Author: Khalilul Afwan, e-mail: khalilafwan@gmail.com.

Published: August , 2025

ABSTRAK

Layanan customer service berperan penting dalam perbankan sebagai penghubung antara nasabah dan institusi keuangan. Di Bank Nagari, layanan ini masih didominasi metode konvensional seperti kunjungan langsung atau sambungan telepon yang memiliki keterbatasan waktu operasional dan rentan antrean panjang. Untuk mengatasi permasalahan tersebut, penelitian ini mengembangkan chatbot multimodal berbasis teks dan suara yang dapat diakses kapan saja untuk memberikan informasi layanan bank secara cepat dan interaktif. Pengembangan sistem menggunakan metode Prototyping dengan tiga tahap utama: Listen to Customer, Build/Revise Mock-up, dan Customer Test Drives Mock-up. Sistem dirancang dengan integrasi React.js (frontend), Go + Gin (backend), AWS Transcribe (speech-to-text), Google Text-to-Speech, serta modul NLP berbasis SBERT dan Random Forest. Hasil pengujian menunjukkan model memiliki akurasi keseluruhan 96,26%, dengan nilai precision, recall, dan f1-score rata-rata 0,96. Sistem mampu mengenali intent pengguna dan memberikan respons otomatis baik dari input teks maupun suara, serta memenuhi kriteria kemudahan penggunaan, kecepatan respons, dan akurasi pemahaman konteks. Implementasi chatbot ini berpotensi meningkatkan kualitas layanan nasabah Bank Nagari secara signifikan.

Keywords: Chatbot Multimodal, Layanan Perbankan, Pemrosesan Bahasa Alami, Pengenalan Suara, SBERT, Random Forest.

ABSTRACT

Customer service plays a crucial role in banking as a communication bridge between customers and financial institutions. At Bank Nagari, customer service is still dominated by conventional methods such as in-person visits and telephone calls, which are limited by operating hours and prone to long queues. To address these issues, this study developed a multimodal chatbot based on text and voice, accessible at any time to provide bank service information quickly and interactively. The system was developed using the Prototyping method with three main stages: Listen to Customer, Build/Revise Mock-up, and Customer Test Drives Mock-up. The architecture integrates React.js (frontend), Go + Gin (backend), AWS Transcribe (speech-to-text), Google Text-to-Speech, and an NLP module based on SBERT and Random Forest. Testing results show an overall accuracy of 96.26%, with average precision, recall, and f1-score of 0.96. The system successfully recognizes user intents and delivers automated responses from both text and voice inputs, meeting functional criteria such as ease of use, fast response time, and accurate context understanding. This multimodal chatbot has the potential to significantly enhance customer service quality at Bank Nagari.

Keywords: Multimodal Chatbot, Banking Services, Natural Language Processing, Speech Recognition, SBERT, Random Forest.

© The Author(s). 2021 Open Access This article is licensed under a Creative Commons Atibution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0

PENDAHULUAN

Customer service merupakan salah satu komponen vital dalam industri perbankan karena berfungsi sebagai penghubung langsung antara nasabah dan institusi keuangan. Perannya mencakup berbagai aktivitas, mulai dari penyampaian informasi produk, pemberian bantuan teknis, penanganan keluhan, hingga penyelesaian transaksi tertentu yang memerlukan verifikasi identitas nasabah (Yulindasari et al., 2022). Kecepatan dan ketepatan layanan menjadi faktor utama yang memengaruhi tingkat kepuasan, kepercayaan, dan loyalitas nasabah terhadap bank. Semakin responsif dan akurat layanan yang diberikan, semakin besar pula kemungkinan nasabah untuk mempertahankan hubungan jangka panjang dengan institusi tersebut.

Di Bank Nagari, mekanisme layanan *customer service* saat ini masih didominasi metode konvensional seperti kunjungan langsung ke kantor cabang atau komunikasi melalui sambungan telepon. Hasil observasi menunjukkan bahwa banyak nasabah terpaksa mengantre cukup lama hanya untuk memperoleh informasi dasar, seperti persyaratan pembukaan rekening, prosedur layanan, atau informasi terkait layanan yang tersedia. Layanan melalui telepon pun belum sepenuhnya mengatasi permasalahan tersebut karena tetap memiliki keterbatasan jam operasional dan sering menghadapi antrean panjang. Kondisi ini menyulitkan nasabah dengan mobilitas tinggi maupun yang berdomisili di daerah terpencil, sehingga kebutuhan akan alternatif layanan yang lebih fleksibel menjadi semakin mendesak.

Perkembangan teknologi digital telah membuka peluang adopsi solusi berbasis kecerdasan buatan (*Artificial Intelligence*,) untuk menjawab tantangan tersebut. Salah satu teknologi yang mulai banyak diimplementasikan adalah *chatbot*, yaitu program komputer yang dirancang untuk berinteraksi dengan pengguna secara otomatis melalui percakapan berbasis teks maupun suara (Adamopoulou & Moussiades, 2020). Dengan keunggulan akses 24 jam, respons instan, dan biaya operasional yang relatif rendah, *chatbot* dapat membantu mengurangi beban layanan konvensional dan meningkatkan efisiensi (Bălan, 2023). Sejumlah penelitian menunjukkan efektivitas teknologi ini. Misalnya, Suparman (2024) melaporkan bahwa penerapan *chatbot* dapat mengurangi waktu tunggu layanan hingga 50%, sementara Mi Alnaser et al. (2023) menekankan peningkatan kepuasan pengguna melalui integrasi AI, meskipun tantangan akurasi dan relevansi respons masih menjadi perhatian.

Dalam pengembangan *chatbot*, teknologi *Natural Language Processing* (NLP) menjadi komponen inti yang memungkinkan sistem memahami maksud (*intent*) dari pertanyaan atau pernyataan pengguna (Park et al., 2022). NLP berfungsi mengubah bahasa manusia menjadi bentuk yang dapat diproses komputer, sehingga respons yang dihasilkan lebih relevan dan kontekstual. Untuk meningkatkan akurasi pengenalan intent, penelitian ini menggabungkan *Sentence-BERT* (SBERT) sebagai metode representasi teks dengan algoritma *Random Forest* sebagai model klasifikasi. SBERT mampu menghasilkan representasi vektor kalimat yang mempertahankan makna kontekstual, sedangkan *Random Forest* memanfaatkan representasi tersebut untuk melakukan klasifikasi dengan tingkat ketelitian tinggi (Assayed et al., 2023).

Selain itu, untuk menghadirkan pengalaman interaksi yang lebih inklusif, sistem juga dilengkapi teknologi *speech-to-text* dan *text-to-speech*. Dengan demikian, pengguna dapat berkomunikasi tidak hanya melalui teks, tetapi juga melalui perintah suara, dan menerima jawaban dalam bentuk audio. Integrasi antara pemrosesan bahasa alami, pengenalan suara, dan sintesis ucapan ini diharapkan dapat memberikan pengalaman layanan nasabah yang lebih praktis, cepat, dan sesuai dengan tuntutan era digital.

METODE PENELITIAN

Penelitian ini menggunakan metode *Prototyping*, yaitu salah satu pendekatan pengembangan perangkat lunak yang berfokus pada pembuatan model awal sistem (*prototype*) untuk diperlihatkan kepada pengguna sehingga dapat diperoleh umpan balik secara cepat dan berulang (Zailani et al., 2020). Pendekatan ini bertujuan memahami kebutuhan pengguna secara lebih mendalam sebelum sistem dikembangkan secara penuh. Metode ini sangat sesuai digunakan pada situasi di mana kebutuhan pengguna belum sepenuhnya terdefinisi di tahap awal pengembangan.

Jenis *prototyping* yang digunakan adalah Prototipe Iterasi Cepat (*Rapid Iterative Prototyping*). Pendekatan ini mengedepankan siklus iteratif yang melibatkan pembuatan, pengujian, serta penyempurnaan *prototype* berdasarkan masukan dari pengguna. Setiap iterasi memberikan kesempatan bagi pengembang untuk melakukan perbaikan dan penyesuaian sehingga sistem yang dihasilkan lebih tepat sasaran, sesuai dengan kebutuhan dan ekspektasi pengguna. Keunggulan metode ini terletak pada kemampuannya mempercepat validasi desain dan fungsi sistem, sekaligus meminimalkan risiko kegagalan karena fitur-fitur utama telah diuji sejak tahap awal pengembangan.
Metode prototype terdiri dari beberapa tahapan utama, yaitu (Hikmah Maulida et al., 2022):

1. *Listen to Customer*

   Pada tahap awal ini dilakukan pengumpulan dan analisis kebutuhan (*requirement gathering and analysis*) melalui tinjau pustaka dan observasi. Data yang diperoleh menjadi dasar dalam menyusun rancangan awal sistem. Setelah itu dilakukan *quick design*, yaitu pembuatan desain cepat yang merepresentasikan pemahaman awal terhadap kebutuhan pengguna. Desain ini dapat berupa sketsa alur proses, diagram alur (*flowchart*), atau ilustrasi sederhana yang mudah dipahami pengguna.
2. *Build or Revise Mock*-up

   Tahap ini mencakup dua kegiatan utama. Pertama, *build prototype*, yaitu pengembangan rancangan awal menjadi bentuk prototipe yang lebih konkret. *Prototype* yang dibuat dapat meliputi diagram UML (misalnya *use case diagram* dan *activity diagram*), *flowchart* sistem, desain basis data, hingga *mock-up* antarmuka pengguna. Kedua, *user evaluation*, yaitu presentasi *prototype* kepada pengguna untuk mendapatkan evaluasi terkait kelebihan, kekurangan, dan potensi perbaikan sistem. Masukan ini menjadi dasar dalam melakukan revisi pada iterasi berikutnya..
3. *Customer Test Drives Mock-up*

   Tahap ini diawali dengan *refining prototype*, yaitu penyempurnaan prototipe berdasarkan masukan yang diperoleh pada tahap sebelumnya. Proses ini dapat dilakukan secara berulang hingga kebutuhan pengguna benar-benar terpenuhi. Setelah prototipe akhir disetujui, tahap dilanjutkan ke *engineer product*, yakni pengembangan sistem secara penuh berdasarkan prototipe akhir, serta penerapan pemeliharaan sistem untuk menjaga kinerja dan stabilitas operasionalnya.

Melalui penerapan metode ini, proses pengembangan sistem chatbot multimodal pada penelitian ini berlangsung secara adaptif, terstruktur, dan terarah. Setiap tahapan pengembangan dilakukan dengan melibatkan pengguna secara langsung, mulai dari tahap perancangan awal hingga pengujian prototype, sehingga masukan yang diberikan dapat segera diintegrasikan ke dalam sistem pada iterasi berikutnya. Pendekatan ini tidak hanya memastikan bahwa fitur dan fungsi yang dibangun selaras dengan kebutuhan nyata di lapangan, tetapi juga membantu mengidentifikasi potensi permasalahan sejak dini. Dengan demikian, kualitas akhir sistem dapat ditingkatkan secara signifikan, baik dari sisi ketepatan fungsi maupun kenyamanan penggunaan, sekaligus meminimalkan risiko terjadinya revisi besar setelah sistem diimplementasikan.

HASIL DAN PEMBAHASAN

HASIL

1. Listen to Customer

Tahap ini diawali dengan proses pengumpulan data dan analisis kebutuhan sistem chatbot berbasis teks dan suara. Pengumpulan data dilakukan untuk mendapatkan informasi yang diperlukan dalam pengembangan sistem chatbot.

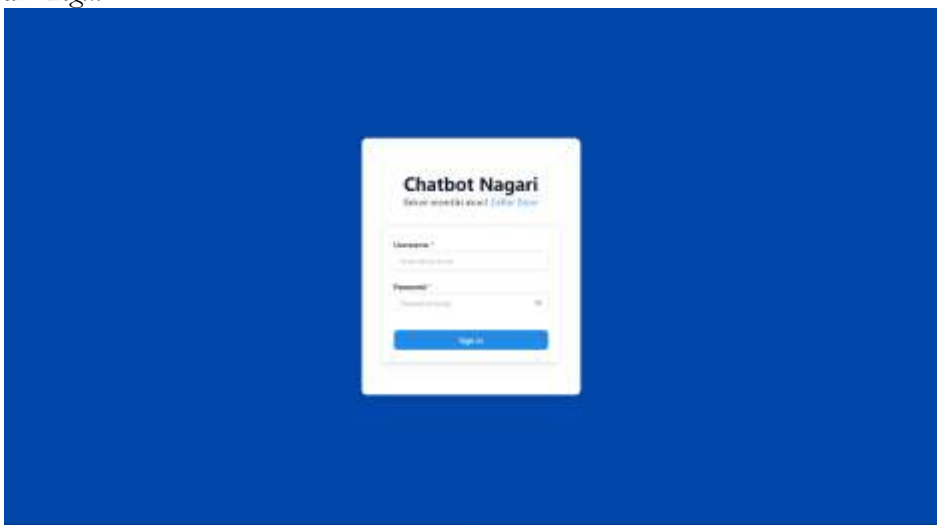
Data dikumpulkan melalui dua cara, yaitu: Observasi dengan cara mengamati sistem layanan pelanggan yang sudah ada untuk mengetahui tantangan dan kekurangan yang dapat diperbaiki dengan chatbot. Selain itu, data juga diperoleh dari Studi Literatur dengan melakukan kajian terhadap berbagai referensi, seperti jurnal, buku, dan penelitian terdahulu yang relevan dengan pengembangan chatbot berbasis NLP. Studi ini bertujuan untuk memahami konsep, teknologi, serta pendekatan terbaik dalam merancang chatbot yang efektif..

2. *Build or Revise Mock-up*

Berdasarkan kebutuhan pengguna yang telah dihimpun pada tahap sebelumnya, dirancanglah prototipe awal sistem penjadwalan dan bimbingan tugas akhir. *Prototype* ini dikembangkan untuk setiap jenis pengguna (*role*), yaitu admin, dosen, dan mahasiswa. Adapun antarmuka utama yang dirancang meliputi:

a. Interface

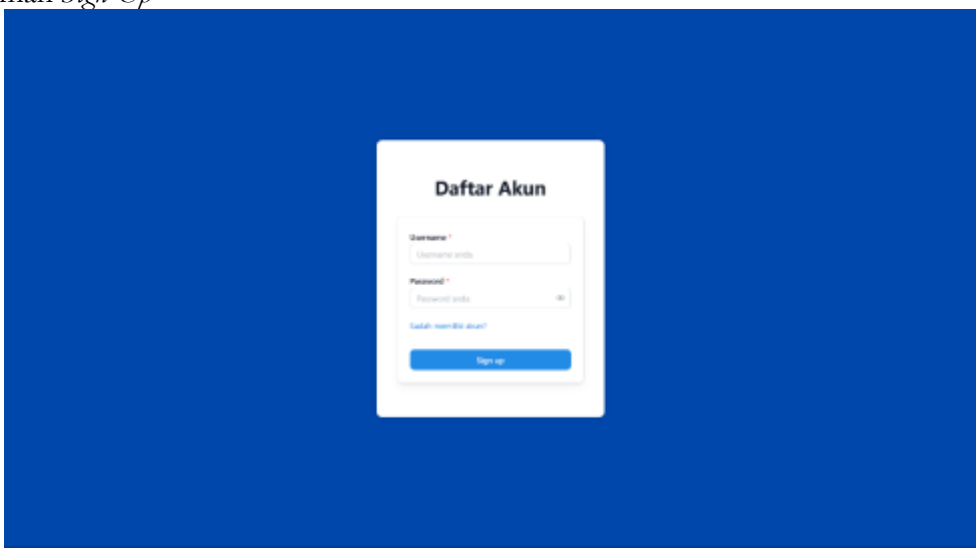
1) Halaman *Login*



Gambar 1. Halaman *Login*

Halaman ini dirancang agar penggun dapat masuk ke sistem untuk bisa melakukan obrolan dengan *chatbot*. Setiap pengguna harus memasukkan kredensial berupa username dan password yang telah terdaftar. Sistem ini akan memverifikasi data dan mengarahkan pengguna ke halaman chat.

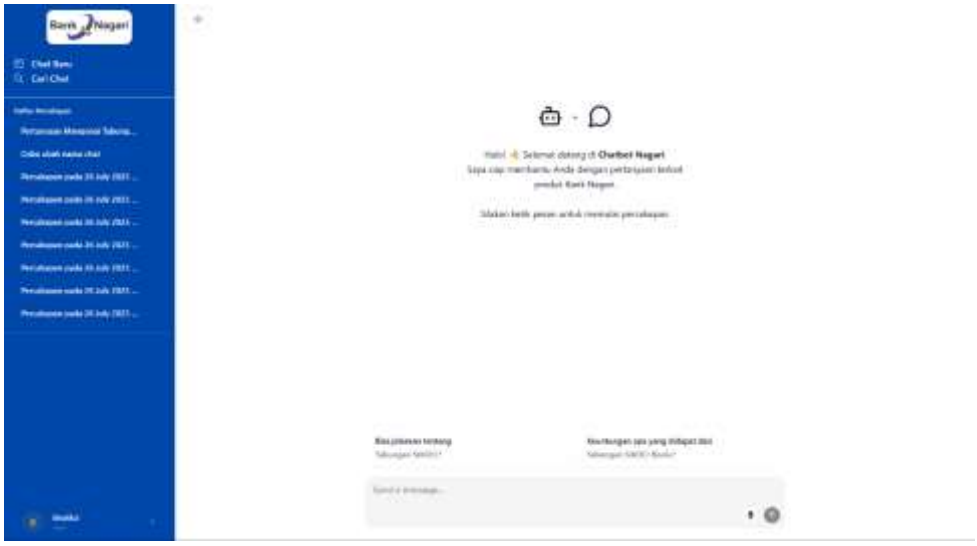
2) Halaman *Sign Up*



Gambar 2. Halaman *Sign Up*

Halaman ini dirancang sebagai pintu awal interaksi pengguna dengan chatbot, untuk mengumpulkan informasi dasar seperti identitas, preferensi bahasa, dan metode interaksi (teks atau suara). Data ini digunakan sistem untuk menyesuaikan alur percakapan agar respons lebih relevan. Desain sederhana dan intuitif dilengkapi panduan serta konfirmasi visual guna mempercepat pendaftaran. Prototipe halaman diuji langsung oleh pengguna untuk menilai kemudahan navigasi, kejelasan instruksi, dan efisiensi, yang menjadi dasar perbaikan pada iterasi berikutnya.

3) Halaman *Chat*



Gambar 3. Halaman *Chat*

Halaman ini merupakan antarmuka utama Chatbot Nagari setelah pengguna *login*. Di tengah layar terdapat sapaan dan kolom input untuk mengajukan pertanyaan terkait produk/layanan Bank Nagari, dengan respons cepat dan efisien. Panel kiri menampilkan riwayat percakapan untuk meninjau kembali atau memulai chat baru. Di bagian bawah tersedia saran pertanyaan cepat agar pengguna bisa memulai percakapan tanpa mengetik dari awal.

b. Proses Logika Sistem

Proses logika sistem pada platform percakapan ini terbagi menjadi beberapa tahap yang saling terhubung, mulai dari interaksi awal pengguna di frontend hingga respons akhir yang ditampilkan kembali ke pengguna. Setiap tahap memiliki peran penting untuk memastikan percakapan berjalan secara real-time, relevan, dan terdokumentasi dengan baik.

1) Tahap pertama dimulai pada sisi frontend, yaitu antarmuka pengguna. Pada tahap ini, pengguna memasukkan teks melalui kolom input yang tersedia di aplikasi. Setelah menekan tombol kirim, teks tersebut dikemas dalam format JSON dan dikirimkan ke backend melalui endpoint API percakapan. Proses ini menjadi jembatan awal interaksi antara pengguna dan sistem, memastikan bahwa data terkirim secara aman dan sesuai format ke server untuk diproses lebih lanjut.
2) Setelah backend menerima data dari frontend, sistem akan meneruskan informasi tersebut ke layanan Natural Language Processing (NLP) yang dikembangkan menggunakan framework Flask. Pengiriman dilakukan menggunakan metode HTTP POST, di mana data teks dikirim sebagai payload ke server Flask. Tahap ini berperan sebagai penghubung antara backend utama dan subsistem NLP, sehingga masukan pengguna dapat dianalisis lebih dalam menggunakan model kecerdasan buatan.
3) Layanan Flask akan memproses masukan teks pengguna melalui beberapa tahapan berurutan, yaitu:
   a) Preprocessing Teks: meliputi penghapusan karakter spesial, konversi seluruh teks ke huruf kecil, dan penghapusan kata umum (stopwords), sehingga data menjadi bersih dan siap diolah.

```
stemmer = StemmerFactory().create_stemmer()
stop_words = set(stopwords.words("indonesian"))

sym_spell = SymSpell(max_dictionary_edit_distance=2,
prefix_length=7)
dictionary_path = pkg_resources.resource_filename(
    "symspellpy", "frequency_dictionary_en_82_765.txt"
)
sym_spell.load_dictionary(dictionary_path, term_index=0,
```

```
count_index=1)

def preprocess_for_sbert(text: str) -> str:
    text = text.lower()
    text = re.sub(r"\s+", " ", text)
    text = re.sub(r"[^\w\s]", "", text)
    return text.strip()

def correct_spelling(text: str) -> str:
    suggestions = sym_spell.lookup_compound(text,
max_edit_distance=2)
    return suggestions[0].term if suggestions else text
```

b) Representasi Vektor dengan SBERT: teks yang telah diproses diubah menjadi sentence embedding menggunakan Sentence-BERT, sehingga makna kontekstual kalimat dapat dipahami oleh sistem.

```
sbert_model = SentenceTransformer('paraphrase-multilingual-
MiniLM-L12-v2')

texts = [preprocess_for_sbert(item['text']) for item in
training_data]
labels = [intent_labels.index(item['intent']) for item in
training_data]

print("□ Mengubah teks menjadi embedding dengan SBERT...")
embeddings = sbert_model.encode(texts, show_progress_bar=True)
```

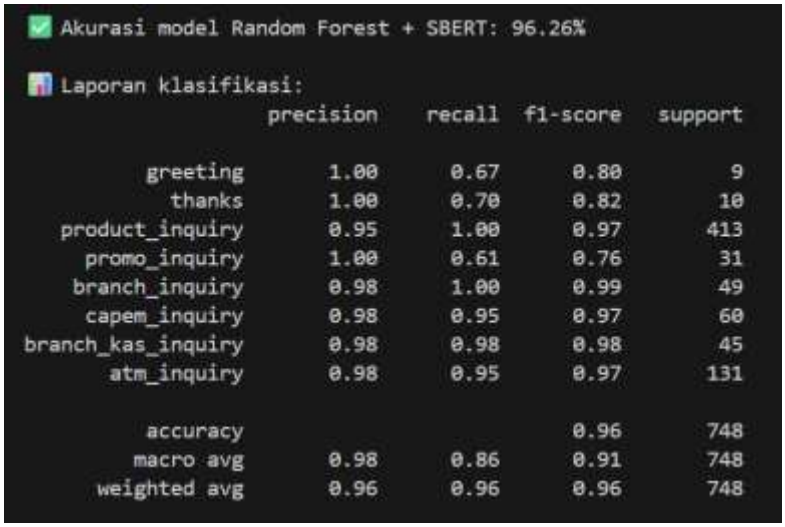
c) Klasifikasi Intent: representasi vektor tersebut dimasukkan ke model Random Forest yang telah dilatih sebelumnya untuk mengidentifikasi intent atau maksud pengguna. Output dari proses ini adalah label intent yang menggambarkan tujuan atau topik dari pertanyaan pengguna.

```
X_train, X_test, y_train, y_test = train_test_split(
    embeddings, labels, test_size=0.2, random_state=42,
stratify=labels
)

print("□ Melatih Random Forest Classifier...")
model = RandomForestClassifier(n_estimators=200,
random_state=42)
model.fit(X_train, y_train)
```

4) Hasil klasifikasi intent dari layanan Flask dikirim kembali ke backend. Backend kemudian melakukan pencocokan intent tersebut dengan template respons yang tersedia di knowledge base. Tujuan tahap ini adalah menemukan respons yang paling relevan berdasarkan peta hubungan antara berbagai jenis intent dan jawaban yang telah disiapkan sebelumnya.
5) Tahap akhir dilakukan di sisi backend, di mana sistem menyusun objek message yang berisi teks balasan, metadata, dan informasi tambahan lainnya. Objek ini kemudian dikirim kembali ke frontend untuk ditampilkan kepada pengguna. Selain itu, seluruh riwayat percakapan, baik masukan pengguna maupun keluaran sistem, disimpan dalam database untuk kepentingan rekam jejak dan analisis di masa mendatang.

c. Hasil Klasifikasi *Intent*



✅ Akurasi model Random Forest + SBERT: 96.26%

📊 Laporan klasifikasi:

	precision	recall	f1-score	support
greeting	1.00	0.67	0.80	9
thanks	1.00	0.70	0.82	10
product_inquiry	0.95	1.00	0.97	413
promo_inquiry	1.00	0.61	0.76	31
branch_inquiry	0.98	1.00	0.99	49
capem_inquiry	0.98	0.95	0.97	60
branch_kas_inquiry	0.98	0.98	0.98	45
atm_inquiry	0.98	0.95	0.97	131
accuracy			0.96	748
macro avg	0.98	0.86	0.91	748
weighted avg	0.96	0.96	0.96	748

Gambar 4. Hasil Klasifikasi menggunakan *Random Forest* + SBERT

Gambar 4. memperlihatkan hasil klasifikasi menggunakan kombinasi *Random Forest* dan SBERT. Model menunjukkan tingkat akurasi keseluruhan sebesar 96,26%, menandakan performa yang sangat baik dalam mengenali berbagai jenis *intent*. Nilai *precision* berada pada kisaran 0,95–1,00, yang berarti prediksi yang dihasilkan sebagian besar akurat. Nilai *recall* bervariasi antara 0,61–1,00, menunjukkan kemampuan model dalam mendeteksi sampel positif di sebagian besar kelas cukup tinggi, meskipun ada perbedaan kinerja antar kategori.

Sebagian besar kelas *intent* memiliki nilai *f1-score* di atas 0,95, menandakan keseimbangan yang baik antara *precision* dan *recall*. Namun, kelas *greeting* dan *thanks* memiliki nilai *recall* yang lebih rendah dibanding kelas lain, disebabkan oleh jumlah dan variasi data yang relatif terbatas pada kategori tersebut. Hal ini membuat model sedikit kurang optimal dalam menangkap seluruh kemungkinan variasi ekspresi pada *intent* tersebut.

Rata-rata tertimbang (*weighted average*) untuk *precision*, *recall*, dan *f1-score* masing-masing sebesar 0,96, 0,96, dan 0,96. Nilai ini menunjukkan bahwa model memiliki kinerja yang konsisten dan andal pada sebagian besar kategori *intent*, sehingga mampu memberikan dasar yang kuat bagi sistem dalam menghasilkan respons yang relevan dan kontekstual pada tahap selanjutnya.

3. Customer Test Drives Mock-up

Pada tahap ini, dilakukan pengujian fungsionalitas dan akurasi untuk memastikan integrasi dan kinerja sistem *chatbot multimodal* berjalan sesuai harapan. Pengujian fungsionalitas dilakukan dengan metode *contract testing* menggunakan pustaka *Pact* untuk memverifikasi komunikasi antara *frontend React* dan *backend Go* pada tiga layanan utama: autentikasi, *chat*, dan *voice*. Hasil pengujian menunjukkan seluruh skenario pada ketiga layanan berhasil dijalankan dengan status "PASS", meliputi *login/logout*, pengelolaan percakapan, hingga unggah pesan suara. Setiap permintaan HTTP dikenali oleh *Pact Mock Server*, dicocokkan dengan kontrak, dan menghasilkan respons sesuai ekspektasi, menandakan bahwa integrasi antar layanan berjalan stabil, konsisten, dan sesuai spesifikasi.

Selain itu, dilakukan pengujian akurasi untuk menilai kemampuan kombinasi algoritma SBERT dan *Random Forest* dalam mengenali *intent* pengguna. Hasilnya menunjukkan akurasi keseluruhan 96,26%, dengan nilai *precision* 0,95–1,00 dan *recall* 0,61–1,00. Sebagian besar kelas *intent* memiliki *f1-score* di atas 0,95, kecuali kelas *greeting* dan *thanks* yang sedikit lebih rendah akibat variasi data terbatas. Rata-rata tertimbang *precision*, *recall*, dan *f1-score* masing-masing sebesar 0,96,

mengindikasikan performa model yang konsisten dan andal dalam memahami maksud pengguna.

PEMBAHASAN

Proses percakapan dimulai dari sisi frontend, tempat pengguna memasukkan teks melalui kolom input aplikasi. Setelah tombol kirim ditekan, data dikemas dalam format JSON dan dikirimkan ke backend melalui API endpoint. Tahap ini berfungsi menjembatani interaksi awal dengan memastikan input pengguna terkirim secara aman dan sesuai format. Setibanya di backend, permintaan diteruskan ke layanan NLP berbasis Flask menggunakan metode HTTP POST. Layanan ini menjadi penghubung antara backend utama dan subsistem pemrosesan bahasa alami.

Di dalam layanan Flask, masukan teks melewati tahap preprocessing, seperti penghapusan karakter khusus, konversi huruf kecil, dan penghilangan stopwords untuk menghasilkan data yang bersih. Teks yang telah diproses kemudian diubah menjadi representasi vektor menggunakan Sentence-BERT (SBERT) yang mempertahankan makna kontekstual. Representasi ini dikirim ke model Random Forest yang sudah dilatih untuk mengidentifikasi intent atau maksud pengguna. Output klasifikasi berupa label intent dikirim kembali ke backend, yang kemudian mencocokkannya dengan template response dalam knowledge base agar sistem dapat memberikan jawaban relevan sesuai konteks.

Tahap akhir adalah penyusunan respons di backend menjadi objek pesan berisi teks balasan, metadata, dan informasi tambahan lain yang diperlukan. Objek ini dikirim ke frontend untuk ditampilkan kepada pengguna secara real-time. Selain itu, seluruh percakapan—baik masukan pengguna maupun keluaran sistem—disimpan di database sebagai rekam jejak, yang dapat dimanfaatkan untuk analisis performa maupun pengembangan fitur di masa mendatang. Dengan alur ini, sistem tidak hanya memberikan jawaban cepat, tetapi juga menjaga kesinambungan data untuk peningkatan layanan..

KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan *chatbot multimodal* berbasis teks dan suara untuk layanan Bank Nagari dengan mengintegrasikan *React.js* sebagai *frontend*, *Go + Gin* sebagai *backend*, AWS *Transcribe* untuk *speech-to-text*, *Google Text-to-Speech*, serta modul NLP berbasis SBERT dan *Random Forest*. Sistem mampu menerima input teks maupun audio, mengenali intent pengguna, dan memberikan respons otomatis sesuai skenario percakapan. Pengujian menunjukkan kombinasi SBERT dan *Random Forest* memiliki akurasi tinggi, didukung metrik *accuracy*, *precision*, *recall*, dan *f1-score* yang konsisten, sehingga efektif dalam memahami konteks percakapan. Penggunaan *MongoDB* sebagai basis data percakapan juga memberikan fleksibilitas penyimpanan teks dan metadata audio. Secara keseluruhan, sistem memenuhi kriteria fungsional berupa kemudahan penggunaan, kecepatan respons, serta ketepatan pemahaman terhadap intent pengguna, sehingga berpotensi meningkatkan kualitas layanan nasabah secara signifikan.

DAFTAR PUSTAKA

Adamopoulou, E., & Moussiades, L. (2020). An Overview of Chatbot Technology. *IFIP Advances in Information and Communication Technology*, *584 IFIP*, 373–383. https://doi.org/10.1007/978-3-030-49186-4_31

Assayed, S. K., Shaalan, K., & Alkhatib, M. (2023). A Chatbot Intent Classifier for Supporting High School Students. *EAI Endorsed Transactions on Scalable Information Systems*, *10*(3). https://doi.org/10.4108/eetsis.v10i2.2948

Bălan, C. (2023). Chatbots and Voice Assistants: Digital Transformers of the Company–Customer Interface—A Systematic Review of the Business Research Literature. *Journal of Theoretical and Applied Electronic Commerce Research*, *18*(2), 995–1019. https://doi.org/10.3390/jtaer18020051

Hikmah Maulida, N., Tanjung Nyaho Jl Yos Sudarso, K., & Raya, P. (2022). Studi Literatur Penerapan Metode Prototype dan Waterfall Dalam Pembuatan Sebuah Aplikasi atau Website. *ResearchGate*.

Mi Alnaser, F., Rahi, S., Alghizzawi, M., & Ngah, A. H. (2023). Does Artificial Intelligence (AI) Boost Digital Banking User Satisfaction? Integration of Expectation Confirmation Model and Antecedents of Artificial Intelligence Enabled Digital Banking. *Heliyon*, *9*(8). https://doi.org/10.1016/j.heliyon.2023.e18930

Park, D. M., Jeong, S. S., & Seo, Y. S. (2022). Systematic Review on Chatbot Techniques and Applications. *Journal of Information Processing Systems*, *18*(1), 26–47. https://doi.org/10.3745/JIPS.04.0232

Suparman, A. (2024). Penerapan Kecerdasan Buatan dalam Sistem Informasi untuk Meningkatkan Pengalaman Pengguna pada Aplikasi E-Commerce. *Journal of Mandalika Literature*, *6*(1), 2745–5963.

Yulindasari, N., Choiriyah, Fadilla, & Tinggi Ekonomi dan Bisnis Syariah, S. (2022). *Strategi Customer Service Dalam Mengatasi Keluhan Nasabah Di Bank Sumsel Babel Cabang Lahat*. *1*(2).