

IMPLEMENTASI RULE-BASED SYSTEM DAN NATURAL LANGUAGE PROCESSING PADA CHATBOT LAYANAN JASA (STUDI KASUS WEBSITE PAPERTRUST)

Abdulah Haris, Shofa Shofiah Hilabi, Elfina Novalia

Sistem Informasi, Universitas Buana Perjuangan Karawang

Jl. H.S. Ronggowaluyo Telukjambe Timur, Karawang-41361, Indonesia

si22.abdulahharis@mhs.ubpkarawang.ac.id

ABSTRAK

Perkembangan teknologi digital di bidang akademik semakin pesat, memunculkan urgensi bagi platform layanan untuk menyediakan respon informasi yang cepat, akurat, dan tersedia secara konstan. Artikel ini membahas pengembangan *chatbot* layanan akademik pada platform *PaperTrust* yang fokus pada verifikasi plagiarisme dan penulisan artikel ilmiah profesional. Masalah utama yang diidentifikasi adalah dependensi pada layanan manual administrasi yang berisiko menurunkan efisiensi operasional dan membatasi aksesibilitas di luar jam kerja. Penelitian ini bertujuan untuk mengidentifikasi kebutuhan, merancang, dan mengimplementasikan *chatbot* berarsitektur *hybrid* yang mengorkestrasikan *Rule-Based System* (RBS) dengan *Natural Language Processing* (NLP) menggunakan metode *Prototype*. Pendekatan ini diperkuat dengan algoritma komputasi *Fuzzy String Matching* menggunakan metode *Levenshtein Distance* untuk menangani variasi dan kesalahan ejaan (*typo*) pada input pengguna. Selain itu, sistem inovasi ini mencakup mekanisme *Web Scraping* untuk pembaruan data secara otomatis, serta modul *Semi-Auto Learning* untuk mengidentifikasi pertanyaan baru. Lebih lanjut, sistem ini mengintegrasikan manajemen memori jangka pendek melalui variabel *session* untuk mendeteksi niat pengguna (agent maksud) dan mengeksekusi transisi kontekstual kepada agen manusia secara proaktif. Pengujian sistem fokus pada evaluasi algoritma menggunakan metode *Black Box Testing* berdasarkan skenario interaksi. Hasil pengujian menunjukkan bahwa integrasi prapemrosesan NLP dan *Levenshtein Distance* secara signifikan mengeliminasi gaya RBS konvensional, di mana sistem berhasil memproses berbagai variasi input pengguna dari 18 skenario pengujian yang dilakukan, seluruh fungsi utama sistem berhasil berjalan sesuai harapan. Secara keseluruhan, sistem *chatbot* mandiri (*native*) ini terbukti berfungsi dengan sangat optimal, sementara fitur pelacakan sesi menyebabkan layanan transisi yang berkelanjutan dan efisien dari mesin ke admin.

Kata kunci : Chatbot, Sistem Berbasis Aturan, Pemrosesan Bahasa Alami, Jarak Levenshtein, Blackbox Testing.

1. PENDAHULUAN

Perkembangan teknologi digital dalam ekosistem akademik semakin mendorong platform layanan jasa untuk menyediakan respons informasi yang cepat, akurat, dan tersedia secara konstan [1]. *PaperTrust* hadir sebagai platform digital yang melayani mahasiswa, dosen, dan peneliti dalam kebutuhan verifikasi plagiarisme menggunakan Turnitin, pengerjaan artikel ilmiah profesional, serta konsultasi akademik berbasis kebutuhan pengguna [2]. Sebagai platform yang terus berkembang, *PaperTrust* memerlukan sistem layanan yang mampu menjawab permintaan pengguna secara efisien tanpa batasan waktu operasional [3].

Masalah utama yang diidentifikasi pada *PaperTrust* adalah ketergantungan pada layanan manual melalui *WhatsApp* dan email yang berisiko menimbulkan respons antrean, menurunkan kepuasan pengguna, dan membatasi aksesibilitas di luar jam operasional [4]. Kondisi ini diperparah dengan banyaknya pertanyaan yang berulang mengenai harga layanan, estimasi waktu pengerjaan, metode pembayaran, dan keamanan dokumen yang seharusnya dapat dijawab secara otomatis oleh sistem. Manurung Menyatakan bahwa keterbatasan layanan manual dapat menurunkan efisiensi operasional dan

berdampak negatif terhadap pengalaman pengguna secara keseluruhan [3].

Berbagai penelitian telah mengembangkan *chatbot* untuk mengatasi permasalahan serupa. Saputra Mengembangkan *chatbot* berbasis *Rule-Based System* untuk layanan akademik, namun tanpa kemampuan menangani kesalahan pengetikan pengguna [5]. Akbar Nasution Mengimplementasikan NLP pada layanan pelanggan *chatbot*, tetapi tanpa mekanisme transmisi ke admin secara proaktif [6],[7]. Mengintegrasikan *Rule-Based System* dengan *Large Language Model* menggunakan platform *Flowise AI* dan *Google Gemini*, namun sistem ini bergantung pada API eksternal berbayar yang memiliki batasan penggunaan. Selain itu, seluruh penelitian tersebut belum mengintegrasikan mekanisme inovasi berbasis pengetahuan *chatbot* secara semi-otomatis.

Berdasarkan kajian terhadap penelitian-penelitian terdahulu, terdapat empat kelemahan utama yang belum terselesaikan secara terpadu: pertama, sebagian besar sistem *chatbot* berbasis *Rule-Based System* tidak memiliki kemampuan menangani kesalahan pengetikan pengguna; kedua, belum ada sistem yang mengintegrasikan mekanisme pengalihan layanan ke admin secara proaktif berdasarkan deteksi niat pengguna; ketiga, dasar pengetahuan *chatbot* bersifat statistik dan harus diperbarui secara manual;

serta keempat, sistem yang umumnya bergantung pada platform atau API eksternal berbayar yang membatasi gangguan dan menambah biaya operasional.

Untuk mengisi keselarasan tersebut, penelitian ini mengembangkan sistem *chatbot* layanan jasa berbasis arsitektur *hybrid* yang mengorkestrasikan *Rule-Based System* dengan *Natural Language Processing* menggunakan metode *Prototype*. Sistem dibangun secara mandiri menggunakan *PHP native* tanpa ketergantungan pada API atau platform eksternal berbayar. Pendekatan ini diperkuat dengan algoritma *Levenshtein Distance* untuk menangani kesalahan pengetikan, *Session Tracking* untuk mendeteksi niat pengguna dan melakukan *Contextual handover* ke admin secara proaktif, *Web Scraping* untuk pembaruan layanan data secara otomatis tanpa akses langsung ke *database*, serta modul *Semi-Auto Learning* berbasis *Human-in-the-Loop* untuk mendeteksi dan mengklasifikasikan pertanyaan yang belum terjawab sebagai kandidat FAQ baru. Kombinasi fitur-fitur tersebut belum ditemukan secara terpadu pada penelitian sejenis sebelumnya.

2. TINJAUAN PUSTAKA

2.1. Chatbot

Chatbot adalah agen percakapan yang mampu memahami dan memproses input bahasa manusia secara otomatis serta memberikan respon yang sesuai [8],[9]. *Chatbot* mengotomatisasi tugas rutin seperti FAQ (*Frequently Asked Questions*), mengurangi beban admin, dan menyediakan layanan 24 jam yang memungkinkan pengguna mendapatkan informasi kapan saja tanpa batasan jam operasional [10], [11].

Pendekatan ini sangat ideal untuk PaperTrust karena menjamin respons cepat pada domain layanan akademik terstruktur.

2.2. Natural Language Processing (NLP)

NLP memungkinkan komputer untuk memproses dan menganalisis bahasa manusia dengan berbagai teknik seperti *tokenisasi*, *stemming*, dan pengenalan entitas [12].

Dalam implementasi *chatbot*, NLP berfungsi untuk mengekstrak makna dari input teks agar dapat diterjemahkan menjadi tindakan atau respons yang tepat oleh sistem [13].

Pada penelitian ini, simulasi NLP diterapkan melalui teknik *stopwords removal* (penghapusan kata umum) untuk meningkatkan fokus pencarian kata kunci inti.

2.3. Fuzzy Matching & Session

Untuk mengatasi kesalahan penulisan (typo) oleh pengguna, digunakan metode *Fuzzy String Matching*. Algoritma *Levenshtein Distance* memungkinkan *chatbot* mengenali kata kunci meskipun pengguna salah mengetikkannya. Selanjutnya, sesi memungkinkan aplikasi *web* untuk mempertahankan status pengguna saat berpindah interaksi [14].

Session digunakan sebagai memori jangka pendek untuk menyimpan riwayat percakapan yang mendasari fitur *Contextual Handover*.

2.4. Rule-Based System (RBS)

Rule-Based System lebih sesuai diterapkan pada kasus percakapan yang terstruktur dan domain tertutup, seperti layanan akademik, karena mampu menjamin akurasi jawaban informasi yang tinggi sesuai dengan basis pengetahuan yang diprogram, dibandingkan dengan model generatif yang memiliki risiko halusinasi [15].

2.5. Teknologi Web

Sistem antarmuka *chatbot* dibangun memanfaatkan HTML untuk kerangka dialog dan CSS untuk visualisasi yang responsif [16].

JavaScript berfungsi untuk mentransfer data secara asinkron (menggunakan *Fetch API*) [17].

Sementara di sisi *backend*, bahasa pemrograman PHP digunakan sebagai mesin inferensi logika yang memproses pencocokan pola dan algoritma jarak kata [18].

2.6. Penelitian Terkait

Beberapa penelitian sebelumnya telah menjadi acuan dalam pengembangan *chatbot* pada layanan akademik dan jasa. Alvin dkk. mengimplementasikan *chatbot* akademik menggunakan kombinasi *Large Language Model* (LLM) dan *Rule-Based System* (RBS), dengan hasil menunjukkan bahwa RBS lebih efektif dalam menangani pertanyaan terstruktur karena mampu menghindari risiko halusinasi informasi pada model generatif [15],[19].

Selanjutnya Saputra dkk. mengembangkan *chatbot* berbasis WhatsApp dengan pendekatan *Rule-Based System* yang terbukti mampu meningkatkan efisiensi kerahasiaan informasi secara real-time [19].

Penelitian lain oleh Nasution dkk. menerapkan teknik *Natural Language Processing* dalam layanan pelanggan *chatbot*, yang berhasil mengurangi beban kerja serta waktu respon admin [20].

Selain itu, Ananta dkk. menggunakan metode pencocokan pola berdasarkan aturan dalam pengenalan objek wisata, yang mampu meningkatkan efisiensi pelayanan. Namun, pendekatan tersebut masih memiliki keterbatasan dalam menangani variasi bahasa yang tidak terstruktur [21].

Hal serupa juga ditemukan pada penelitian Putra dkk., di mana *chatbot* berbasis *Rule-Based System* mampu mempercepat respons layanan, namun masih terbatas dalam memahami konteks percakapan yang lebih kompleks [18].

Berdasarkan kajian tersebut, dapat disimpulkan bahwa metode *Rule-Based System* efektif untuk layanan dengan domain terbatas, namun memiliki kelemahan dalam menangani variasi input pengguna. Oleh karena itu, penelitian ini mengintegrasikan pendekatan *Rule-Based System* dengan teknik *Natural Language Processing* dan algoritma *Levenshtein*

Distance untuk meningkatkan toleransi terhadap kesalahan pengetikan, serta memanfaatkan pelacakan sesi untuk mendukung interaksi yang lebih kontekstual dan proaktif.

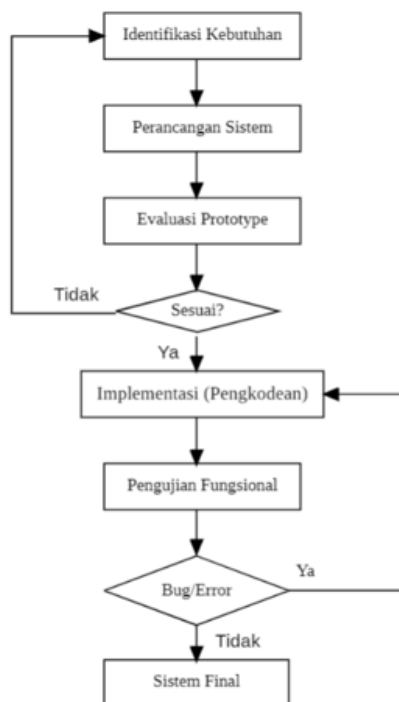
3. METODE PENELITIAN

Penelitian ini menggunakan metode Prototype sebagai pendekatan pengembangan perangkat lunak. Metode ini dipilih karena memungkinkan proses pengembangan dilakukan secara bertahap melalui komunikasi intensif dengan pengguna, sehingga kebutuhan sistem dapat diidentifikasi dan disempurnakan pada setiap iterasi.

Tahapan penelitian meliputi analisis kebutuhan, perancangan sistem menggunakan Unified Modeling Language (UML), implementasi chatbot berbasis Rule-Based System dan Natural Language Processing (NLP), serta pengujian sistem menggunakan Black Box Testing untuk seluruh fungsi berjalan sesuai kebutuhan fungsional.

3.1. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode *Prototype*[22]. Metode ini dipilih karena sesuai untuk pengembangan sistem interaktif yang membutuhkan umpan balik secara cepat dan berulang.



Gambar 1. Metode Prototype

Berikut adalah penjelasan dari setiap tahapan dalam alur metode *Prototype* yang diterapkan pada penelitian ini:

a. Identifikasi Kebutuhan

Tahap ini dilakukan melalui observasi langsung terhadap website *PaperTrust* dan wawancara dengan owner *PaperTrust* untuk mengidentifikasi

permasalahan layanan manual yang ada, pertanyaan yang sering diajukan *pengguna* (*Frequently Asked Questions*), serta fitur-fitur yang dibutuhkan sistem *chatbot*. Hasil identifikasi dirangkum dalam tabel kebutuhan sistem sebagai acuan seluruh tahapan pengembangan berikutnya.

b. Perancangan Sistem

Berdasarkan hasil identifikasi kebutuhan, dilakukan perancangan sistem yang meliputi pembuatan *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, serta desain antarmuka pengguna *chatbot*. Perancangan ini berfungsi sebagai blueprint sistem sebelum masuk ke tahap implementasi.

c. Evaluasi Prototype

Hasil perancangan sistem dievaluasi bersama owner *PaperTrust* untuk memastikan desain yang dibuat telah sesuai dengan kebutuhan yang diidentifikasi sebelumnya. Evaluasi ini berfokus pada kesesuaian fitur dan alur percakapan *chatbot*, bukan pada pengujian teknis sistem. Jika desain dinilai kurang sesuai, dilakukan revisi perancangan sebelum masuk ke tahap implementasi.

d. Implementasi (Pengkodean)

Sistem *chatbot* dibangun menggunakan bahasa pemrograman *PHP* secara *native* tanpa ketergantungan pada *API* atau platform eksternal berbayar. Implementasi mencakup modul *NLP Preprocessing*, algoritma *Levenshtein Distance* untuk penanganan *typo*, *Session Tracking* untuk *contextual handover*, untuk pembaruan data layanan secara otomatis, serta modul *Semi-Auto Learning* untuk mendeteksi dan mengklasifikasikan pertanyaan yang belum terjawab.

e. Pengujian Fungsional

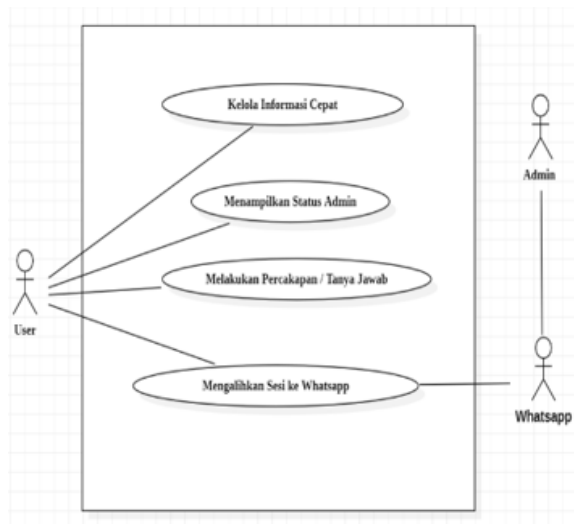
Pengujian dilakukan oleh peneliti menggunakan metode *Black Box Testing* setelah sistem selesai diimplementasikan. Berbeda dengan evaluasi *Prototype* yang menilai kesesuaian desain, pengujian fungsional berfokus pada kebenaran teknis sistem, yaitu apakah setiap komponen dan skenario interaksi menghasilkan output yang sesuai harapan. Jika ditemukan bug atau error, sistem dikembalikan ke tahap implementasi untuk diperbaiki dan diuji kembali.

f. Sistem Final

Setelah seluruh pengujian fungsional dinyatakan berhasil tanpa bug, sistem *chatbot* "Nixi" dinyatakan final dan aktif digunakan pada website *PaperTrust* sebagai layanan informasi otomatis yang beroperasi selama 24 jam penuh.

3.2. Use Case Diagram

Use Case Diagram menggambarkan interaksi antara aktor dengan sistem *chatbot*. Terdapat dua aktor utama dalam sistem ini, yaitu Pengguna dan Admin *PaperTrust*. *Use Case Diagram* sistem *chatbot* *PaperTrust* ditunjukkan pada Gambar 2 berikut.

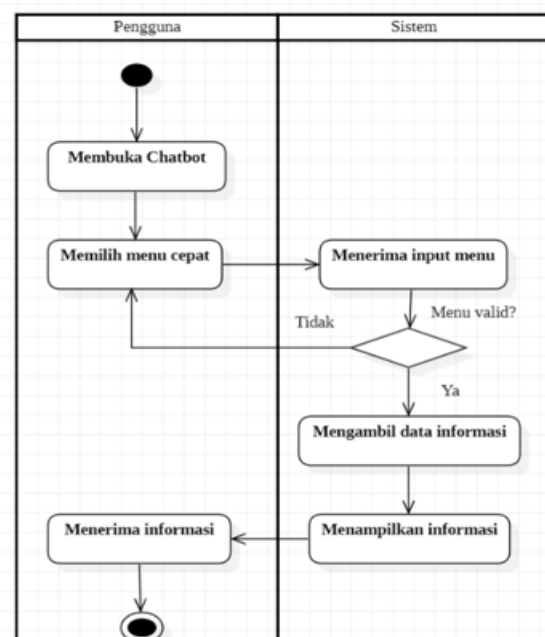


Gambar 2. Use Case Diagram

Gambar 2 menunjukkan *Use Case Diagram* chatbot pada platform *PaperTrust*. Sistem melibatkan dua aktor utama, yaitu Pengguna dan Admin. Pengguna dapat mengakses informasi melalui fitur Kelola Informasi Cepat, melakukan percakapan dengan chatbot, melihat status admin, serta mengalihkan sesi ke *WhatsApp* jika diperlukan.

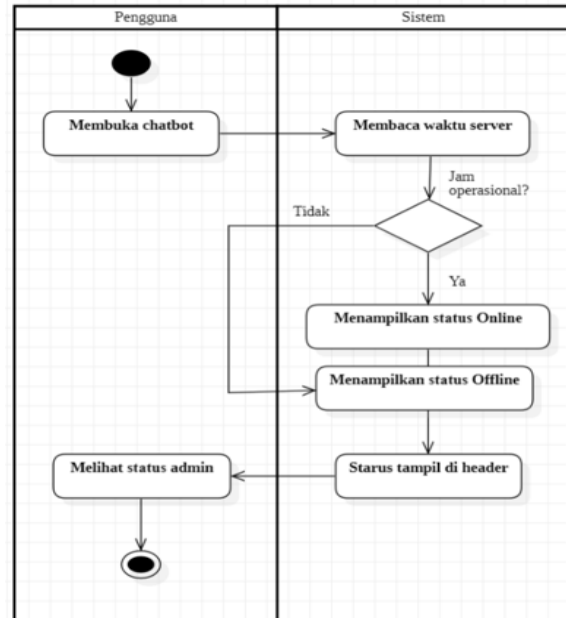
3.3. Activity Diagram

Activity Diagram digunakan untuk menggambarkan alur aktivitas sistem berdasarkan setiap fungsi utama yang terdapat pada *Use Case Diagram*. Pada penelitian ini, *Activity Diagram* terdiri atas empat diagram yang merepresentasikan proses Kelola Informasi Cepat, Menampilkan Status Admin, Melakukan Percakapan/Tanya Jawab, dan Mengalihkan Sesi ke *WhatsApp*.



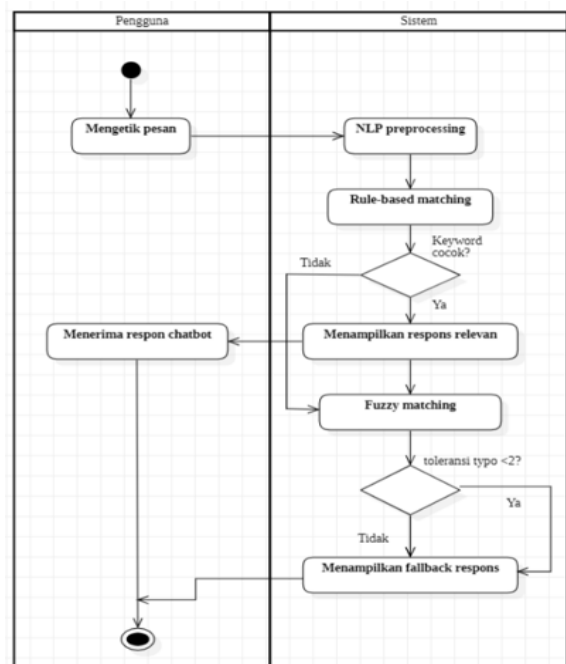
Gambar 3. Aktiviti diagram kelola informasi cepat

Gambar 3 menunjukkan alur aktivitas fitur Kelola Informasi Cepat. Proses dimulai ketika pengguna memilih menu informasi dengan cepat, kemudian sistem memproses permintaan dan menampilkan informasi layanan yang tersedia kepada pengguna.



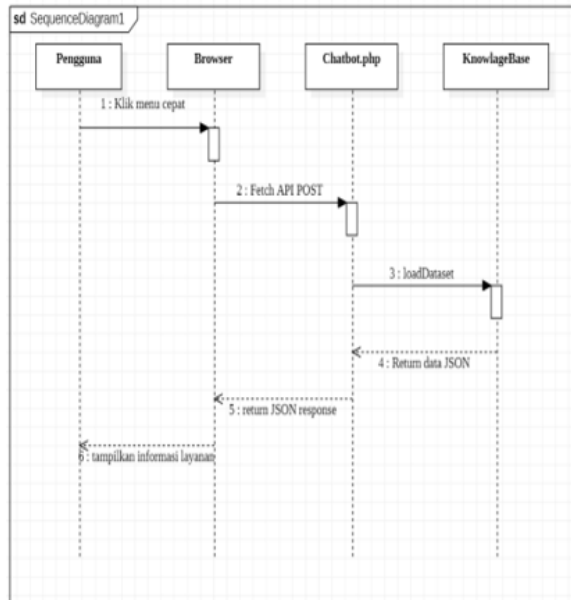
Gambar 4. Activity Diagram Menampilkan Status Admin

Gambar 4 menunjukkan alur aktivitas fitur Menampilkan Status Admin. Sistem menerima permintaan pengguna, mengambil status admin secara *real-time*, kemudian menampilkan informasi ketersediaan admin kepada pengguna.



Gambar 5. Activity diagram Kelola Pesan/Percakapan

Gambar 5 menunjukkan alur aktivitas proses percakapan antara pengguna dan *chatbot*. Sistem menerima masukan pengguna, melakukan NLP *Preprocessing*, *Rule-Based Matching*, serta *Levenshtein Distance* sebelum menghasilkan respons yang sesuai.



Gambar 6. Sequence Diagram Kelola Informasi Cepat

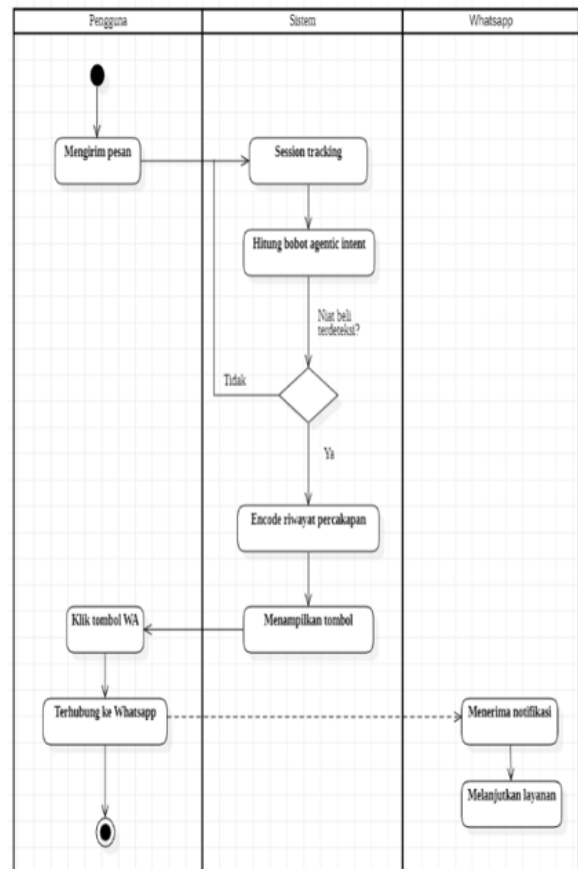
Gambar 6 menunjukkan alur aktivitas sesi percakapan ke *WhatsApp*. Proses dilakukan ketika *chatbot* mendeteksi bahwa permintaan pengguna memerlukan admin sehingga riwayat percakapan dipertahankan untuk melanjutkan proses pelayanan. Alur dimulai dari proses *session Tracking* yang merekam percakapan percakapan pengguna menggunakan *PHP \$_SESSION*. Sistem secara terus-menerus menghitung bobot agen berdasarkan pola interaksi pengguna, seperti menanyakan harga, estimasi waktu pengerjaan, dan penggunaan kata-kata pemesanan.

3.4. Sequence Diagram

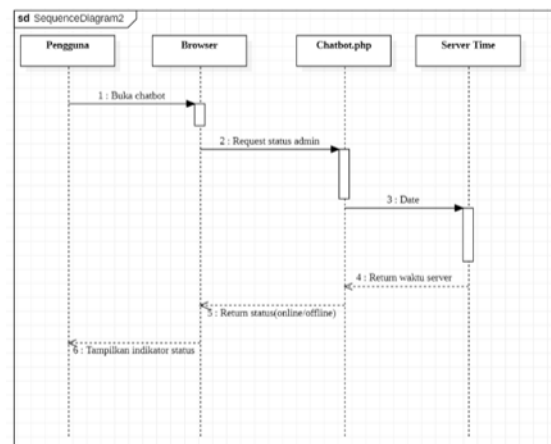
Sequence Diagram menggambarkan urutan interaksi antar komponen sistem *chatbot PaperTrust* secara kronologis. Komponen yang terlibat meliputi *Browser (JavaScript/Fetch API)*, *Backend PHP (chatbot.PHP)*, modul *NLP Preprocessor*, *Matching Engine*, *Session Manager*, *KnowledgeBase (JSON Dataset)*, serta aktor eksternal *Admin/WhatsApp*. Sesuai dengan empat *Use Case* utama, terdapat empat *Sequence Diagram* yang dijelaskan sebagai berikut.

Gambar 7 menunjukkan urutan interaksi antara pengguna, antarmuka chatbot, dan backend dalam proses mengelola informasi dengan cepat. Permintaan pengguna diproses dengan memuat data layanan dan FAQ, kemudian sistem menghasilkan respons berdasarkan menu yang dipilih serta memuat informasi status admin. Respons tersebut selanjutnya

ditampilkan kepada pengguna melalui antarmuka chatbot.

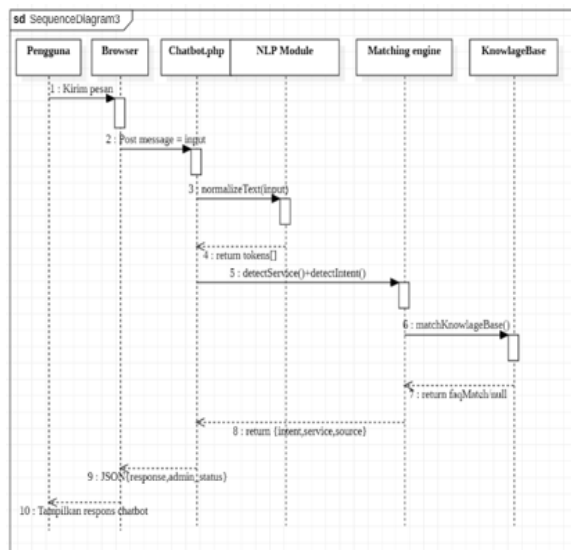


Gambar 7. Activity diagram mengarahkan ke Whatsapp



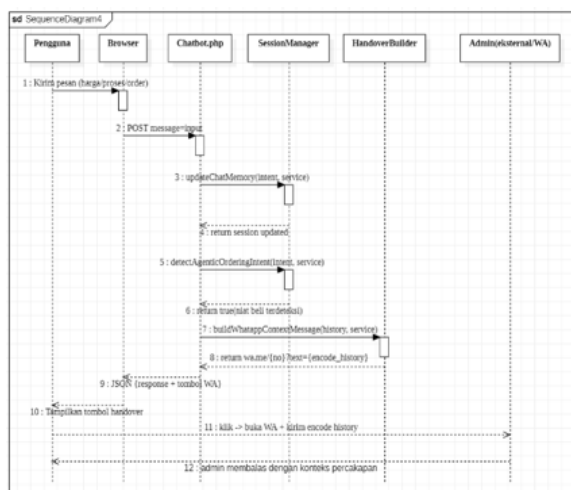
Gambar 8. Sequence Diagram Menampilkan Status Admin

Gambar 8 menunjukkan proses penampilan status admin pada *chatbot*. Sistem pemeriksaan waktu operasional berdasarkan konfigurasi *server*, kemudian menghasilkan informasi status admin yang dikirim bersama respon *chatbot*. Informasi tersebut ditampilkan secara *real-time* pada antarmuka pengguna.



Gambar 9. Sequence Diagram Kelola Pesan/Percakapan

Gambar 9 menunjukkan urutan interaksi dalam proses pengelolaan percakapan antara pengguna dan *chatbot*. Proses diawali ketika pengguna mengirimkan pesan, kemudian sistem melakukan NLP *Preprocessing*, pencocokan pertanyaan menggunakan *Rule-Based System* dan algoritma *Levenshtein Distance*, serta menyusun respon yang sesuai. Selanjutnya, sistem memperbarui sesi percakapan, mencatat riwayat interaksi, dan mengirimkan respons kepada pengguna melalui antarmuka *chatbot*.

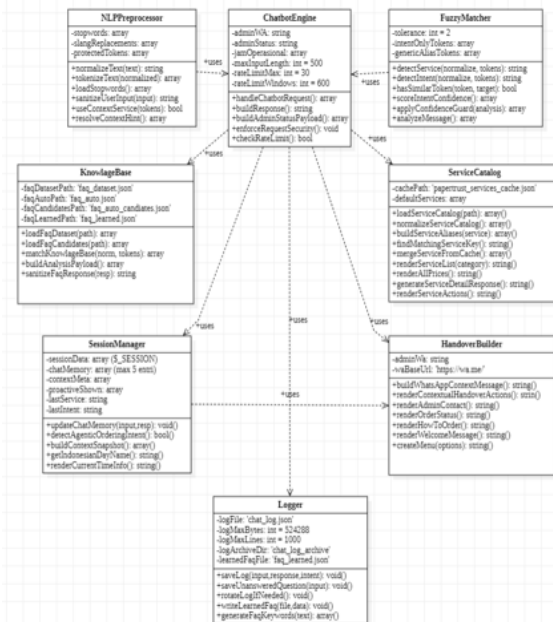


Gambar 10. Sequence Diagram Mengarahkan ke WhatsApp

Gambar 10 menunjukkan urutan interaksi dalam proses percakapan ke *WhatsApp*. Proses diawali ketika sistem mendeteksi bahwa pengguna memerlukan layanan lanjutan berdasarkan konteks percakapan yang telah tersimpan. Selanjutnya, sistem menyusun ringkasan percakapan dan menghasilkan tautan *WhatsApp* sehingga pengguna dapat melanjutkan komunikasi dengan admin tanpa mengulang informasi yang telah disampaikan.

3.5. Class Diagram

Class Diagram menggambarkan struktur kelas-kelas utama yang membangun sistem *chatbot PaperTrust* beserta hubungan antar kelas tersebut. Kelas-kelas utama dalam sistem ini adalah *ChatbotEngine*, *NLPPreprocessor*, *FuzzyMatcher*, *SessionManager*, *KnowledgeBase*, *ServiceCatalog*, *HandoverBuilder*, *Logger* dan *AutoLearning*. *Class Diagram* sistem *chatbot PaperTrust* ditunjukkan pada Gambar 11 berikut.



Gambar 11. Class Diagram

Class Diagram pada Gambar 11 menunjukkan struktur kelas yang membangun sistem *chatbot PaperTrust* beserta hubungan antar modul. Sistem terdiri atas modul *ChatbotEngine*, *NLPPreprocessor*, *FuzzyMatcher*, *KnowledgeBase*, *ServiceCatalog*, *SessionManager*, *HandoverBuilder*, dan *Logger* yang saling berkolaborasi dalam memproses masukan pengguna, mengelola basis pengetahuan, mempertahankan konteks percakapan, melakukan handover kontekstual, serta mencatat aktivitas sistem. Struktur ini menggambarkan pembagian tanggung jawab setiap modul sehingga pengembangan dan pemeliharaan sistem dapat dilakukan secara lebih terstruktur.

3.6. Implementasi dan Modul Sistem

Sistem *chatbot* dibangun secara *native* menggunakan bahasa pemrograman *PHP* yang terintegrasi ke dalam antarmuka situs web *PaperTrust*. Implementasi sistem ini dibagi ke dalam beberapa modul utama yang saling terintegrasi:

3.7. Dataset PAQ dan Prapemrosesan NLP

Dasar pengetahuan *chatbot* disusun dalam format *JSON* (*FAQ_dataset.json*) yang memuat daftar kata kunci dan respons layanan *PaperTrust*. Sebelum dilakukan pencocokan aturan, setiap input pengguna dinormalisasi melalui modul NLP *Preprocessing* yang

mengeksekusi tiga tahapan: *Case Folding* (konversi huruf kapital), penghapusan tanda baca menggunakan *regular Expression*, dan *Stopword Removal* untuk

Tabel 1. Dataset Chatbot

No	Kata Kunci (Keywords)	Respon Chatbot
1	Aman, Rahasia, Repository, No Repo, Private, Privasi	Chatbot memberikan informasi bahwa dokumen dijamin kerahasiaannya dan layanan Cek Turnitin menggunakan akun No Repository sehingga file tidak tersimpan pada database publik.
2	Waktu, Proses, Lama, Deadline, Cepat, Berapa Hari, Selesai Kapan	Chatbot menjelaskan estimasi waktu pengerjaan layanan, di mana Cek Turnitin bersifat instan, sedangkan layanan Parafrase dan Penulisan disesuaikan dengan tingkat kesulitan dan jumlah halaman.
3	Pembayaran, Transfer, Qris, Bayar, Cara Bayar, Metode Bayar	Chatbot memberikan informasi terkait metode pembayaran yang tersedia, yaitu melalui QRIS, transfer bank, dan e-wallet seperti DANA dan GoPay.

Dataset pada Tabel 2 merupakan contoh basis pengetahuan *chatbot* yang disusun dalam bentuk pasangan kata kunci dan respon. Dataset lengkap disimpan dalam format *JSON* dan digunakan sebagai referensi utama sistem *chatbot* dalam menghasilkan respon otomatis terhadap input pengguna.

3.8. Algoritma Rule-Based dan Fuzzy Matching

Sistem menggunakan *Rule-Based Matching* sebagai mesin inferensi utama untuk mencari kecocokan eksak (pencocokan tepat) antara input bersih pengguna dengan *dataset*. Apabila kecocokan eksak tidak ditemukan, sistem mengaktifkan algoritma komputasi *Fuzzy String Matching* menggunakan metode *Levenshtein Distance*. Algoritma ini menghitung jarak perbedaan karakter dengan batas toleransi maksimal dua karakter (*typo_tolerance* = 2) untuk mendeteksi dan menangani kesalahan pengetikan (*typo*) dari pengguna secara dinamis.

3.9. Pelacakan Sesi (Session Tracking)

Sistem mengimplementasikan memori jangka pendek menggunakan fungsi *PHP Session* (*\$_SESSION*) untuk melacak riwayat percakapan pengguna secara *real-time*. Modul ini bertugas menganalisis pola interaksi untuk mendeteksi niat transaksi (*Agentic Intent*). Ketika indikasi pemesanan terdeteksi, sistem secara proaktif mengeksekusi *Contextual handover* dengan memunculkan tombol integrasi ke *WhatsApp* admin yang telah disisipi rangkuman riwayat percakapan (*URL-encoded*), sehingga admin dapat langsung memproses pesanan.

menghapus kata-kata umum yang tidak berarti menggunakan referensi *stopwords.json*.

3.10. Web Scraper dan Pembelajaran Semi-Otomatis

Guna memastikan kebaruan informasi tanpa perlu akses langsung ke *database* utama, sistem menggunakan modul *Web Scraper* (*web_scraper.PHP*) yang mengekstrak informasi layanan dan harga dari website secara otomatis menggunakan fungsi *file_get_contents()* untuk disimpan dalam *cache JSON*. Selain itu, modul *Semi-Auto Learning* menerapkan pendekatan *Human-in-the-Loop* dengan menganalisis riwayat percakapan (*chat_log.json*) guna mengidentifikasi maksud yang gagal dijawab (*fallback*). Pertanyaan-pertanyaan tersebut diajukan berdasarkan bunyi teks untuk menghasilkan kandidat FAQ baru yang siap dijalankan oleh admin.

3.11. Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa seluruh komponen dan fitur sistem *chatbot PaperTrust* berjalan sesuai dengan yang diharapkan. Metode pengujian yang digunakan adalah *Black Box Testing*, yaitu metode pengujian yang berfokus pada fungsionalitas sistem dari perspektif pengguna tanpa memperhatikan struktur kode internal. Pengujian dilakukan oleh peneliti berdasarkan skenario interaksi yang telah dirancang. *Black Box Testing* pada penelitian ini dibagi menjadi dua level pengujian:

3.12. Pengujian Level 1 Komponen Fungsional

Pengujian Level 1 bertujuan untuk menguji setiap komponen sistem secara individual guna memastikan masing-masing komponen berjalan sesuai fungsinya.

Tabel 2. Rancangan Pengujian Black Box Testing Level 1

No	Komponen	Input	Output Diharapkan
1	NLP - Case Folding	"Halo Min Apakah AMAN?"	"halo min apakah aman"
2	NLP - Stopword Removal	"halo min apakah aman dari repo turnitin"	"AMAN REPO TURNITIN"
3	Fuzzy - Typo Ringan (jarak=1)	"parafras"	Terdeteksi sebagai "parafrase"
..	..	..	..
12	Semi-Auto Learning	Eksekusi <i>auto_learning.PHP</i> setelah ada <i>log fallback</i>	<i>FAQ_auto_candidate.json</i> berisi kandidat FAQ baru

3.13. Pengujian Level 2 Skenario Interaksi Pengguna

Pengujian Level 2 bertujuan untuk menguji sistem secara *end-to-end* berdasarkan alur percakapan nyata pengguna, untuk memastikan seluruh komponen bekerja secara terintegrasi dan menghasilkan respons yang tepat dalam konteks interaksi pengguna yang sesungguhnya.

Mulai dari pertanyaan normal, input dengan *typo*, input di luar domain, alur pengguna dengan niat beli, akses di luar jam operasional, hingga pertanyaan menggunakan sinonim kata.

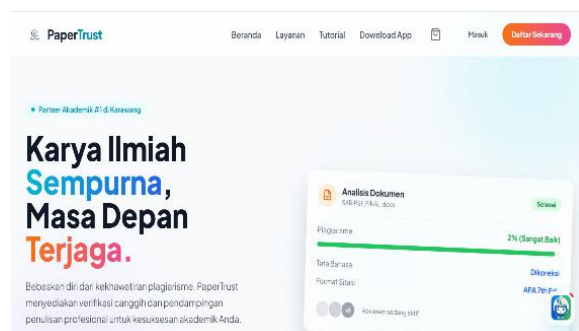
Tabel 3. Rancangan Pengujian Black Box Testing Level 2

No	Skenario	Alur Input Pengguna	Output Diharapkan
1	Pertanyaan normal tanpa typo	"cara bayar"	Info QRIS, transfer bank, dan e-wallet ditampilkan
2	Input dengan typo pada kata kunci	"biaya prfrase brpa?"	Sistem mendeteksi "parafrase", info harga ditampilkan
3	Input di luar domain layanan	"siapa presiden indonesia?"	Respons fallback halus ditampilkan
4	Alur pengguna dengan niat beli	Tanya harga → tanya lama proses → ketik "mau pesan"	Sistem mendeteksi agentic intent, tombol handover WA + history muncul

4. HASIL DAN PEMBAHASAN

4.1. Implementasi Antarmuka Chatbot

Antarmuka *chatbot* "Nixi" diimplementasikan pada situs web *PaperTrust* menggunakan *HTML*, *CSS*, dan *JavaScript*. Sistem ini menggunakan pendekatan asinkron melalui *Fetch API* untuk memungkinkan interaksi waktu nyata tanpa perlu memuat ulang halaman. *Chatbot* diintegrasikan sebagai *widget* mengambang di sudut kanan bawah layar (Gambar 12), memungkinkan pengguna untuk mengakses layanan kapan saja tanpa harus meninggalkan halaman aktif.



Gambar 12. Hasil Penanganan Kesalahan Pengetikan oleh Sistem

4.2. Kemampuan Sistem Menangani Variasi Input Pengguna

Bagian ini menunjukkan efektivitas sistem *chatbot PaperTrust* dalam memproses berbagai variasi input pengguna, yang mencakup kondisi input normal, kesalahan pengetikan (*typo*), serta input di luar layanan domain. Analisis pada modul ini sekaligus menjadi jawaban atas rumusan masalah kedua mekanisme mengenai penanganan pesan pengguna secara efektif dan deterministic.



Gambar 13. Implementasi Chatbot pada Website Papertrust

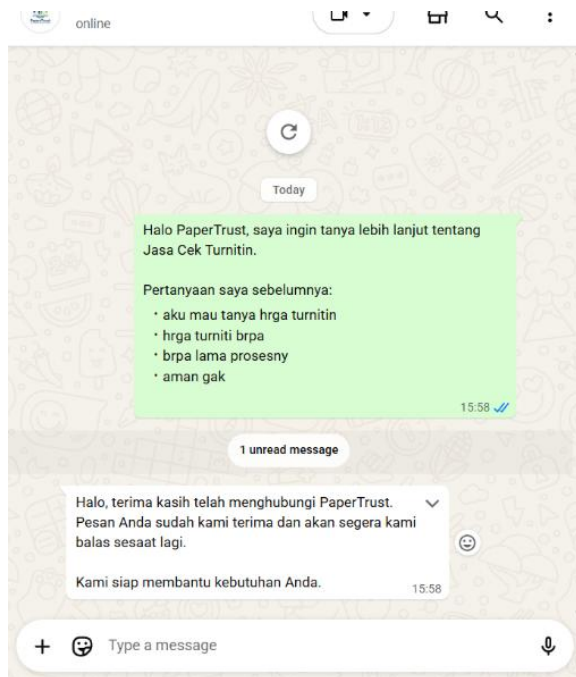
Sebagai contoh, ketika pengguna mengetikkan "biaya prfrase brpa?", sistem berhasil mengidentifikasi bahwa "prfrase" merupakan *typo* dari "parafrase" dengan jarak *Levenshtein* sebesar dua, sehingga sistem tetap dapat memberikan informasi harga jasa Parafrase yang relevan. Gambar 13 menunjukkan kemampuan sistem dalam menangani input dengan kesalahan pengetikan.

4.3. Pengalihan Layanan ke Admin Secara Proaktif

Kemampuan sistem dalam mendeteksi niat pengguna dan melakukan pengalihan layanan ke admin secara proaktif. Kemampuan ini diimplementasikan melalui fitur *Session Tracking* dan *Contextual handover* yang merupakan bagian dari jawaban atas rumusan masalah kedua penelitian ini.

Saat pengguna melakukan interaksi misalnya: menanyakan "harga", lalu "lama proses", dan mengetik kata "pesan" atau "pesanan" algoritma menambahkan bobot pada variabel *Agentic Intent*. Ketika kemungkinan niat.

Gambar 14 menunjukkan tampilan *Contextual handover* pada sistem serta bukti riwayat percakapan yang diterima oleh admin melalui *WhatsApp*. Dengan fitur ini, proses transisi dari layanan otomatis ke layanan admin manusia berlangsung secara mulus dan efisien.



Gambar 14. Hasil Contextual Handover ke Admin WhatsApp

4.4. Hasil Pengujian Blackox Testing

Pengujian fungsional sistem dilakukan menggunakan metode *Black Box Testing* untuk memastikan bahwa seluruh komponen dan fitur sistem *chatbot PaperTrust* berjalan sesuai dengan yang diharapkan. Hasil keseluruhan pengujian *Black Box Testing* Level 1 dan Level 2 ditunjukkan pada Tabel 4 berikut.

Tabel 4. Hasil Pengujian Black Box Testing

Level Pengujian	Total Skenario	Berhasil	Tidak Berhasil	Persentase
Level 1 Pengujian Fungsional	12	12	0	100%
Level 2 Skenario Interaksi Pengguna	6	6	0	100%
Total	18	18	0	100%

Berdasarkan Tabel 4 sistem *chatbot PaperTrust* berhasil mencapai tingkat keberhasilan sebesar 100% dari total 18 skenario pengujian, terdiri dari 12 skenario Level 1 dan 6 skenario Level 2. Tidak ditemukan bug atau error pada seluruh skenario pengujian, sehingga sistem dinyatakan telah memenuhi seluruh kebutuhan fungsional yang ditetapkan dan siap digunakan secara aktif pada Website *PaperTrust*.

Hasil pengujian ini menegaskan bahwa arsitektur *hybrid* yang dibangun secara mandiri menggunakan *PHP native* tanpa ketergantungan *API* eksternal, mampu menghasilkan sistem *chatbot* yang akurat, adaptif, dan Integrasi seluruh komponen sistem terbukti berjalan secara sinergis dan menghasilkan

pengalaman interaksi yang efektif bagi pengguna layanan *PaperTrust*.

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengimplementasikan *chatbot* layanan pada platform *PaperTrust* dengan mengintegrasikan *Rule-Based System*, *Natural Language Processing* (NLP), algoritma *Levenshtein Distance*, serta *Session Tracking* untuk mendukung *Contextual handover* ke admin *WhatsApp*. Integrasi tersebut memungkinkan *chatbot* menangani variasi input pengguna, mengatasi kesalahan pengetikan (*typo*), mempertahankan konteks percakapan, dan memberikan layanan secara lebih proaktif. Hasil *Black Box Testing* menunjukkan bahwa seluruh 18 skenario pengujian berhasil dijalankan sesuai dengan kebutuhan fungsional, sehingga sistem dinyatakan mampu memberikan layanan informasi secara konsisten dan siap diterapkan pada website *PaperTrust*.

Untuk penelitian selanjutnya, disarankan melakukan *User Acceptance Test* (UAT) guna memberikan tingkat kepuasan dan kemudahan penggunaan sistem dari perspektif pengguna. Selain itu, algoritma *Levenshtein Distance* dapat dikembangkan dengan menerapkan batas toleransi yang lebih adaptif sesuai karakteristik panjang kata agar kemampuan sistem dalam menangani variasi input menjadi lebih optimal.

DAFTAR PUSTAKA

- [1] Y. E. Ananta, D. Yuniati, D. Rolliawati, A. Kunaefi, and A. Permadi, "Pengembangan Aplikasi Sido Chatbot sebagai Aplikasi Pengenalan Objek Wisata Kediri Menggunakan Rule-Based Pattern Matching," *Jurnal Teknologi dan Informasi*, vol. 15, no. 1, pp. 40–53, Mar. 2025.
- [2] Novia Sri Falinda, Rosy Febriani Daud, Rizky Rahmanda Irawan, Muhammad Fatchuriz, and Arianto, "Penguatan Integritas Ilmiah di Era Transformasi Digital: Implementasi Reposisi Peran Universitas Swasta melalui Kuliah Umum Internasional sebagai Bentuk Pengabdian kepada Masyarakat," *Jurnal Ragam Pengabdian*, vol. 2, no. 2, pp. 399–412, Aug. 2025.
- [3] J. Manurung *et al.*, "Pengaruh Penerapan Material Requirements Planning (MRP) Terhadap Efisiensi Manajemen Persediaan Pada Perusahaan Manufaktur," 2025.
- [4] A. R. A. Syafaruddin, N. Natsir, and S. Syafaruddin, "Implementasi Sistem Informasi Akuntansi (SIA) Berbasis Cloud Dalam Meningkatkan Efisiensi Operasional Bisnis Kecil," *Jurnal Minfo Polgan*, vol. 13, no. 2, pp. 1618–1626, Oct. 2024.
- [5] Deden Moh Alfiansyah, Wiilys, Lila Setiyani, Devi Fajar Wati, and Dedih, "Pengembangan Chatbot Berbasis Web untuk Layanan Informasi

- di Horizon University,” *bit-Tech*, vol. 7, no. 3, pp. 1068–1077, Apr. 2025.
- [6] S. Ernawati, I. Maulana, and N. Ruhyana, “Simpatik: Jurnal Sistem Informasi dan Informatika Penerapan Natural Language Processing Pada Model Chatbot Hybrid Untuk Optimasi Pelayanan Open Journal Systems.” [Online].
- [7] M. Akbar Nasution *et al.*, “Implementasi NLP Dalam Pembuatan Chatbot Customer Service Publisher Jurnal Studi Kasus LARISMA,” 2024.
- [8] G. Syahrani, S. Sevira, A. Yunizar, and P. Yusuf, “Rancangan Chatbot Rekomendasi Coffee Shop Jabodetabek dengan Menggunakan Dialogflow Natural Language Processing,” vol. 7, no. 1, pp. 74–84, 2024.
- [9] T. Asmala, R. A. Barokah, K. : Rezky, and A. Barokah, “Penerapan Artificial Intelligence Dalam Administrasi Operasional: Studi Kasus Pada Startup di Bidang Jasa Pengiriman Tiket Kuy Di Kota Cimahi,” *Jurnal Akuntansi dan Manajemen Keuangan*, vol. 6, no. 1, 2025.
- [10] Rizal Chandra Rivaldi and T.D. Wismarini, “Analisis Sentimen Pada Ulasan Produk Dengan Metode Natural Language Processing (NLP),” *Elkom: Jurnal Elektronika dan Komputer*, vol. 17, no. 1, pp. 120–128, Jul. 2024.
- [11] M. Erlangga Fauzi and Tata Sutabri, “PublicTalk: Sistem Chatbot Pintar Berbasis Natural Language Processing untuk Layanan Pemerintahan Digital,” *JOURNAL SAINS STUDENT RESEARCH*, vol. 3, no. 2, pp. 426–433, Apr. 2025.
- [12] R. M. Ulah and I. M. Suartana, “Implementasi Session Management Pada Website Magang Menggunakan Teknologi MERN,” *Journal of Informatics and Computer Science*, vol. 06, 2024.
- [13] A. Alvin, R. Robet, and F. A. Tarigan, “Implementasi Chatbot Otomatis Akademik Berbasis Web Menggunakan LLM dan Rule-Based System Studi Kasus: STMIK Time,” *JIKO (Jurnal Informatika dan Komputer)*, vol. 9, no. 3, p. 651, Oct. 2025.
- [14] B. Huda, S. Shofiah Hilabi, and M. Rahayuningsih, “Android Based Employee Absence and Leaving Application Information System,” *Buana Information Technology and Computer Sciences (BIT and CS)*, vol. 2, no. 1, 2021.
- [15] S. Shofiah Hilabi, A. Solehudin, and S. Susanto, “Application for Submission of Research Recommendations and Practice Work Web Based,” *Buana Information Technology and Computer Sciences (BIT and CS)*, vol. 22, no. 1, 2021.
- [16] G. Putu Mahendra Putra, A. Tenriawaru, P. Studi Ilmu Komputer, F. Matematika dan Ilmu Pengetahuan Alam, and U. Halu Oleo, “Rancang Bangun Virtual Assistant Chatbot Menggunakan Node.Js pada Layanan Sistem Informasi Akademik,” 2023.
- [17] F. Saputra, R. M. Handoko, W. Putra, R. Priskila, and V. H. Pranatawijaya, “Chatbot Berbasis Whatsapp Teknik Informatika Universitas Palangkaraya: Rules Based System,” *Jurnal Teknologi Informatika dan Komputer*, vol. 10, no. 1, pp. 296–308, Mar. 2024.
- [18] E. J. A. H. Nasution, L. Molefe, and R. T. Utami, “Platform E-Learning Adaptif Meningkatkan Aksesibilitas bagi Berbagai Demografi Pembelajar,” *Jurnal MENTARI: Manajemen, Pendidikan dan Teknologi Informasi*, vol. 3, no. 2, pp. 177–186, Mar. 2025.
- [19] A. L. Hananto, B. Priyatna, A. Haris, U. Buana, and P. Karawang, “Application of Prototype Method on Student Monitoring System Based on WEB,” 2020.
- [20] E. Novalia and A. Voutama, “Black Box Testing dengan Teknik Equivalence Partitions Pada Aplikasi Android M-Magazine Mading Sekolah,” 2022.