

Implementasi *Cloud Computing* Pada Aplikasi *Chatbot* Sebagai *Virtual Assisten*

Zikra Afnadi¹, Mursyidah^{2*}, Fachri Yanuar Rudi³

^{1,2,3} *Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe
Jln. B.Aceh Medan Km.280 Buketrata 24301 INDONESIA*

¹zikraafnadi.lsm2015@gmail.com

^{2*}mursyidah@pnl.ac.id

¹fachri@pnl.ac.id

Abstrak— Penelitian ini mengimplementasikan cloud computing pada aplikasi chatbot berbasis Artificial Intelligence Markup Language (AIML) untuk Jurusan Teknologi Informasi dan Komputer di Politeknik Negeri Lhokseumawe guna meningkatkan efisiensi dan responsivitas layanan akademik. Hasil uji menunjukkan bahwa responsivitas chatbot masih perlu ditingkatkan, dengan 33,3% responden menyatakan cukup setuju dalam menangani pertanyaan ambigu, dan 25% tidak setuju dan sangat tidak setuju terhadap jawaban yang diberikan. Tingkat kepuasan mahasiswa mencapai 75%. Evaluasi Quality of Service (QoS) menunjukkan performa jaringan yang sangat baik dalam throughput, packet loss, delay dan jitter pada beban tinggi, meskipun terjadi penurunan performa pada throughput di beban rendah. Penelitian ini berkontribusi pada peningkatan efisiensi layanan dan penyediaan aplikasi inovatif bagi mahasiswa.

Kata kunci— Layanan jurusan, Chatbot, Artificial Intelligence, Quality of Service, AIML

Abstract— This research implements cloud computing on an Artificial Intelligence Markup Language (AIML)-based chatbot application for the Department of Information and Computer Technology at Politeknik Negeri Lhokseumawe to improve the efficiency and responsiveness of academic services. The test results show that the responsiveness of the chatbot still needs to be improved, with 33.3% of respondents expressing moderate agreement in handling ambiguous questions, and 25% disagreeing and strongly disagreeing with the answers given. The level of student satisfaction reached 75%. Quality of Service (QoS) evaluation showed excellent network performance in throughput, packet loss, delay and jitter at high load, although there was a drop in performance in throughput at low load. This research contributes to improving service efficiency and providing innovative applications for students.

Keywords— Department services, Chatbot, Artificial Intelligence, Quality of Service, AIML

I. PENDAHULUAN

Pelayanan jurusan dalam konteks pendidikan tinggi merujuk pada rangkaian layanan yang disediakan oleh departemen atau jurusan di sebuah perguruan tinggi. Layanan jurusan memberikan informasi dan bimbingan kepada mahasiswa, terutama mahasiswa baru, seperti menyusun dan menyediakan informasi terkait struktur kurikulum, jadwal perkuliahan, serta memberikan bantuan dalam perencanaan akademis [1].

Pelayanan jurusan juga berperan sebagai salah satu tempat untuk menanggapi pertanyaan terkait perkuliahan. Dalam mengoptimalkan pelayanan jurusan maka dikembangkan aplikasi *chatbot* untuk membantu mahasiswa agar lebih mudah mengakses informasi terkait perkuliahan [2].

Aplikasi *chatbot* menjadi inovasi baru dalam mengembangkan pelayanan di Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe. Aplikasi chatbot menggunakan teknologi *Artificial Intelligence* (AI) dengan metode *Artificial Intelligence Markup Language* (AIML) sebagai dasar pembuatan sistem[3]. Hal ini dilakukan agar aplikasi chatbot dapat memberikan respon yang sesuai dengan kebutuhan pengguna.

A. Pelayanan Jurusan

Pelayanan jurusan di lingkungan pendidikan tinggi merupakan aspek kritis dalam mendukung pengalaman akademis mahasiswa. Pelayanan jurusan mencakup serangkaian layanan yang dirancang untuk memfasilitasi mahasiswa dalam memahami, mengakses, dan mengelola informasi terkait program studi dan kebutuhan akademis [4].

Pelayanan jurusan berperan sebagai sumber informasi utama bagi mahasiswa. Ini mencakup memberikan bimbingan terkait struktur kurikulum, pemilihan mata kuliah, jadwal perkuliahan, persyaratan pendaftaran, serta berbagai informasi lainnya yang mendukung keberhasilan akademis mahasiswa.

B. Artificial Intelligence

Artificial Intelligence atau Kecerdasan Buatan merupakan teknologi yang sangat populer saat ini. *Artificial Intelligence* merujuk pada kemampuan mesin untuk melakukan tugas yang memerlukan kecerdasan manusia. Ini mencakup pemrosesan bahasa alami, pengenalan pola, pemecahan masalah, dan pengambilan keputusan (Supangat dkk., 2021).

Pengembangan *Artificial Intelligence* (AI) telah mengalami evolusi selama beberapa tahun. Awalnya, dominasi sistem berbasis aturan menggunakan logika simbolik, namun terbatas dalam menangani kompleksitas. Berkembangnya ke *Machine Learning* (ML) memungkinkan mesin belajar dari

data tanpa pemrograman eksplisit, dengan algoritma yang mengidentifikasi pola.

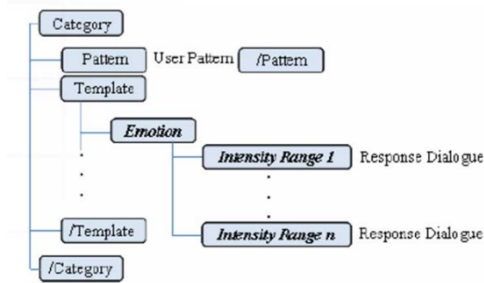
C. *Artificial Intelligence Markup Language (AIML)*

AIML, atau *Artificial Intelligence Markup Language*, adalah bahasa *markup* yang dikhususkan untuk memprogram respons dan perilaku mesin berbasis kecerdasan buatan, terutama digunakan dalam pengembangan *chatbot*. AIML berfokus pada kemampuan komunikasi antara manusia dan mesin dengan menggunakan sintaksis yang terstruktur.

AIML terdiri dari aturan-aturan yang disusun dalam kategori (*category*). Setiap kategori memiliki dua bagian utama: pola kalimat (*pattern*) dan jawaban (*template*) (Muhammad dkk., 2023). Pola kalimat digunakan untuk mencocokkan Input dari pengguna, sedangkan jawaban berisi respons atau tindakan yang diinginkan.

Beberapa struktur AIML yang perlu dipahami dalam pembuatan chatbot, yaitu:

1. *AIML*
Tag utama yang digunakan untuk pembuka dan penutup dokumen AIML.
2. *Category*
Digunakan sebagai acuan chatbot dalam memberikan respons terhadap pola kalimat dari pengguna.
3. *Pattern*
Pola kalimat yang digunakan sebagai acuan untuk menganalisa pesan dari pengguna.
4. *Template*
Jawaban atau respon yang diberikan oleh *chatbot* kepada pengguna. Tag ini dapat berupa teks, gambar, suara, atau kombinasi dari beberapa jenis media



GAMBAR 1. STRUKTUR TAG AIML

TABEL I
POTENSI KONVERSI BEBERAPA RADIONUKLIDA

Nuklida	Energi Neutron		
	Termal	Epitermal	Cepat
Pu-239	1,09	0,9	1,9
U-233	1,20	1,3	1,5
U-235	1,07	0,8	1,3

No	Tags	Keterangan
1	<star>	Digunakan ketika menyematkan dalam tag
2	<srail>	Untuk mencocokkan katagori
3		Digunakan untuk memberika tanggapan acak dari daftar yang diberikan
4	<topic>	Digunakan untuk menyimpan konteks sehingga tanggapan dapat didasarkan pada konteks itu
5	<random>	Digunakan untuk medapatkan tanggapan secara acak
6	<get>	Untuk mengambil suatu nilai dari variabel
7	<set>	Untuk mengunakan nilai dari variabel
8	<think>	Digunakan untk menyimpan variabel
9	<condition>	Digunakan untuk menanggapi <i>Input</i> yang cocok

D. *User Interface & User Experience*

User Interface (UI) adalah jembatan yang memungkinkan pengguna berinteraksi dengan program atau sistem melalui elemen visual seperti bentuk, warna, ikon, dan teks, yang dirancang untuk menarik perhatian dan mempermudah penggunaan. Sementara itu, *User Experience* (UX) berkaitan dengan bagaimana pengalaman pengguna terbentuk selama interaksi dengan produk atau layanan, dan mempengaruhi cara pengguna berinteraksi dengan *product* yang digunakan. Pendekatan yang terstruktur diperlukan untuk mengatasi tantangan terkait UX agar pengalaman pelanggan dapat dikelola secara efektif (Putra dkk., 2021).

Tujuan utama dari UI adalah untuk memfasilitasi interaksi yang efisien dan efektif antara pengguna dan sistem dengan menyediakan elemen desain yang mudah diakses dan digunakan. Berikut adalah beberapa komponen dasar dalam mendesain *User Interface*:

1. *Elemen Visual*
Ini termasuk semua komponen grafis seperti tombol, ikon, dan kontrol yang memungkinkan pengguna untuk berinteraksi dengan sistem.
2. *Desain Interaktif*
Mengacu pada bagaimana elemen-elemen tersebut berfungsi dan berinteraksi dengan pengguna, termasuk efek *hover*, animasi, dan *feedback* visual.
3. *Tata Letak dan Struktur*
Menentukan bagaimana elemen-elemen UI disusun di layar untuk memudahkan navigasi dan penggunaan. Ini mencakup *grid*, spasi, dan hierarki visual.

Selain itu, dalam ilmu *User Interface* juga ada yang dimakan dengan Prinsip Desain UI yang meliputi beberapa hal sebagai berikut:

1. *Konsistensi*
Menjaga keseragaman dalam desain untuk memudahkan pengguna memahami dan mengingat elemen-elemen antarmuka.
2. *Kejelasan*
Menyediakan elemen desain yang jelas dan mudah dipahami untuk mengurangi kebingungan dan meningkatkan efisiensi penggunaan.

3. Aksesibilitas

Memastikan bahwa antarmuka dapat digunakan oleh semua pengguna, termasuk mereka yang memiliki keterbatasan fisik atau kognitif.

Berbeda dengan *User Interface*, tujuan *User Experience* adalah untuk menciptakan pengalaman yang positif dan memuaskan dengan memenuhi kebutuhan dan harapan pengguna, untuk itu UX memiliki beberapa hal yang jadi point utama dalam penerapannya, yaitu:

1. Komponen UX

- Kegunaan atau *Usability*: Mengukur seberapa mudah dan efisien pengguna dapat menyelesaikan tugas-tugas menggunakan sistem. Ini melibatkan aspek seperti navigasi, kejelasan instruksi, dan efektivitas fungsi.

- Fungsionalitas: Menilai sejauh mana sistem memenuhi kebutuhan pengguna dan menyediakan fitur yang berguna.

- Emosi dan Perasaan: Menggali bagaimana pengalaman pengguna memengaruhi perasaan mereka terhadap produk atau layanan, termasuk kepuasan dan frustrasi.

2. Prinsip Desain UX

- Fokus pada Pengguna: Merancang dengan mempertimbangkan kebutuhan, harapan, dan konteks pengguna. Ini melibatkan riset pengguna dan pengujian untuk memahami pengalaman mereka.

- Kesesuaian dengan Tujuan: Memastikan bahwa desain UX mendukung tujuan pengguna dan memfasilitasi pencapaian hasil yang diinginkan dengan mudah.

- Peningkatan Berkelanjutan: Terus-menerus mengumpulkan umpan balik dari pengguna dan melakukan iterasi desain untuk meningkatkan pengalaman secara berkelanjutan.

3. Proses Desain UX

- Riset Pengguna: Mengumpulkan informasi tentang kebutuhan, perilaku, dan masalah pengguna melalui wawancara, survei, dan analisis data.

-Perancangan *Prototype*: Mengembangkan *prototype* yang memungkinkan pengujian dan evaluasi desain sebelum implementasi penuh.

- Pengujian dan Evaluasi: Melakukan uji coba dengan pengguna untuk menilai kegunaan, efektivitas, dan kepuasan terhadap desain yang diusulkan.

UI dan UX adalah dua aspek yang saling terkait namun berbeda dari desain produk dan aplikasi. UI berfokus pada antarmuka visual dan elemen desain yang memfasilitasi interaksi pengguna, sementara UX mencakup keseluruhan pengalaman pengguna dan bagaimana produk memenuhi kebutuhan dan harapan mereka. Kombinasi dari desain UI yang efektif dan UX yang positif adalah kunci untuk menciptakan produk yang sukses dan memuaskan bagi pengguna.

E. Cloud Computing

Cloud computing merupakan teknologi inovatif yang saat ini banyak diperbincangkan oleh para ahli dan pengguna teknologi informasi. Teknologi ini dirancang untuk

memungkinkan akses ke sumber daya dan aplikasi dari berbagai lokasi melalui jaringan internet (Setiawan dkk., 2022). *Cloud computing* terdiri dari tiga komponen utama: karakteristik, model layanan, dan model implementasi.

Karakteristik *cloud computing* mencakup:

1. *On-demand self-service*

Pengguna dapat mengakses dan mengelola layanan sesuai kebutuhan tanpa intervensi manusia.

2. *Broad network access*

Akses yang luas melalui jaringan internet dari berbagai perangkat.

3. *Resource pooling*

Penggabungan sumber daya untuk melayani banyak pengguna dengan cara yang dinamis.

4. *Rapid elasticity*

Kemampuan untuk menyesuaikan kapasitas sumber daya dengan cepat sesuai kebutuhan.

5. *Measured service*

Pengukuran penggunaan sumber daya untuk pembayaran sesuai penggunaan.

Model layanan *cloud computing* terdiri dari:

1. *Software as a Service (SaaS)*

Penyedia layanan menawarkan aplikasi yang dapat diakses melalui internet.

2. *Platform as a Service (PaaS)*

Menyediakan platform untuk pengembangan aplikasi tanpa mengelola infrastruktur.

3. *Infrastructure as a Service (IaaS)*

Menyediakan sumber daya infrastruktur seperti server dan penyimpanan secara virtual.

Model implementasi *cloud computing* meliputi:

1. *Public Cloud*

Layanan cloud yang tersedia untuk umum dan dikelola oleh penyedia pihak ketiga.

2. *Private Cloud*

Layanan *cloud* yang khusus digunakan oleh satu organisasi dengan kontrol penuh terhadap infrastruktur.

3. *Hybrid Cloud*

Kombinasi dari *public* dan *private cloud*, memungkinkan pertukaran data dan aplikasi di antara keduanya.

4. *Community Cloud*

Layanan cloud yang dibagi oleh beberapa organisasi dengan kebutuhan serupa.

Pembagian ini memberikan kerangka kerja untuk memahami berbagai aspek dari teknologi *cloud computing* dan aplikasinya dalam berbagai konteks.

F. Chatbot

Chatbot merupakan program komputer yang dirancang untuk berinteraksi dengan pengguna melalui pesan teks atau suara. Terdapat dua jenis utama *chatbot*, yaitu berbasis aturan dan berbasis kecerdasan buatan. *Chatbot* berbasis aturan mengikuti aturan yang telah ditentukan sebelumnya,

memberikan respons sesuai dengan pola yang telah diprogramkan. Meskipun efektif dalam skenario tertentu, *chatbot* jenis ini mungkin tidak dapat menangani input yang tidak sesuai dengan aturan yang telah ditetapkan. Di sisi lain, *chatbot* berbasis kecerdasan buatan menggunakan teknologi kecerdasan buatan untuk memahami dan memproses bahasa alami pengguna (Kusnanda dkk., 2022).

G. Flutter

Flutter merupakan *framework* pengembangan *User Interface* yang menggunakan bahasa pemrograman *Dart*. *Dart* dirancang untuk mengoptimalkan pengembangan aplikasi mobile dengan memadukan kecepatan eksekusi dan kemudahan pemeliharaan. Konsep utama dalam *Flutter* adalah *widget*, yang merupakan elemen-elemen pembangun UI. *Flutter* menggunakan *widget* untuk membangun antarmuka pengguna dan setiap komponen UI dianggap sebagai *widget*, baik itu komponen sederhana seperti tombol hingga struktur yang lebih kompleks seperti tata letak.

Keunggulan *Flutter* terletak pada kemampuannya untuk membangun aplikasi untuk berbagai platform seperti Android, iOS, web, dan desktop dengan menggunakan kode yang sama. Ini memberikan fleksibilitas yang besar bagi pengembang (Adi dkk., 2022).

H. Python

Python adalah bahasa pemrograman tingkat tinggi yang bersifat *general-purpose*. Dirancang untuk kemudahan pembacaan dan penulisan kode, *Python* memiliki sintaksis yang bersih dan jelas, yang membuatnya sangat cocok untuk pengembangan perangkat lunak. Contoh pustaka terkenal termasuk NumPy, pandas, Flask, Django, TensorFlow, dan PyTorch. *Python* juga dapat dijalankan di berbagai platform dan bersifat open-source, memungkinkan akses dan kontribusi dari komunitas pengembang global. *Python* juga memiliki kemampuan integrasi yang baik dengan bahasa lain seperti C/C++, Javascript dan Java (Aruriansyah dkk., 2023).

I. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah standar industri yang digunakan untuk merancang, memodelkan, dan mendokumentasikan sistem perangkat lunak. UML menyediakan notasi dan pedoman yang konsisten untuk menggambarkan struktur dan perilaku sistem (Mubarak dkk., 2019). Tujuan utama UML adalah menyediakan bahasa umum yang dapat dimengerti oleh semua pemangku kepentingan, seperti analis, pengembang, dan manajer proyek.

J. Black Box

Metode *Black Box Testing* adalah teknik pengujian yang digunakan untuk mengidentifikasi kesalahan dalam sistem aplikasi, seperti kesalahan pada fungsionalitas sistem dan hilangnya menu aplikasi. *Black Box Testing* berfokus pada pengujian aspek fungsionalitas sistem aplikasi tanpa melihat struktur internalnya. Pengujian dilakukan dengan memberikan *input* data acak untuk mendapatkan hasil yang terprediksi. Jika data *input* salah, sistem akan menolaknya atau tidak

menyimpannya ke dalam basis data; sebaliknya, jika data input benar, sistem akan menerima dan menyimpannya ke dalam basis data (Uminingsih dkk., 2022).

Teknik pengujian dalam *Black Box Testing* mencakup berbagai metode, antara lain

- a) *Equivalence Partitioning*, yaitu teknik yang membagi *input* data menjadi beberapa partisi untuk pengujian.
- b) *Boundary Value Analysis*, yang bertujuan untuk mendeteksi kesalahan pada batas nilai minimum dan maksimum, baik di dalam maupun di luar perangkat lunak.
- c) *Fuzzing*, yaitu teknik yang mencari bug atau gangguan dalam perangkat lunak melalui injeksi data yang cacat.
- d) *Cause-Effect Graph*, yang menggunakan grafik untuk menggambarkan hubungan antara penyebab dan efek, serta mengarahkan proses pengujian;
- e) *Orthogonal Array Testing*, digunakan untuk menguji domain input yang relatif kecil tetapi kompleks, dengan fokus pada pengujian skala besar.
- f) *All Pair Testing*, yang mendesain semua pasangan test case untuk mengeksekusi setiap kemungkinan kombinasi diskrit dari parameter input, bertujuan untuk mencakup seluruh pasangan yang mungkin.
- g) *State Transition*, teknik yang digunakan untuk menguji kondisi mesin dan navigasi melalui representasi grafik.

Dalam penelitian ini, pengujian pada aplikasi *chatbot* dilakukan menggunakan teknik *Equivalence Partitioning*. Teknik ini membagi data input dari unit perangkat lunak menjadi beberapa partisi, dari mana *test case* dapat diturunkan. Prinsip utama dari teknik ini adalah merancang *test case* yang mencakup setiap partisi setidaknya satu kali.

II. METODOLOGI PENELITIAN

A. Pengumpulan Data

Pengumpulan data dilakukan langsung dengan mewawancarai beberapa mahasiswa jurusan terkait dengan penelitian ini serta pengamatan pribadi secara langsung selama perkuliahan.

B. Perancangan Metode

Perancangan metode *Artificial Intelligence Markup Language* (AIML) akan diimplementasikan pada aplikasi *chatbot* layanan jurusan. Penerapan metode ini diterapkan untuk meningkatkan kemampuan interaksi dengan pengguna.



Gambar 2. Struktur Tag Aiml

Gambar 2 menggambarkan alur kerja metode *Artificial Intelligence Markup Language* (AIML) dalam konteks pengembangan aplikasi *chatbot* untuk layanan jurusan berbasis AIML. Proses dimulai dari tahap awal *start*, di mana aplikasi *chatbot* menerima input dari pengguna, seperti mahasiswa yang ingin mendapatkan informasi terkait jurusan Teknologi Informasi dan Komputer. Setelah *input* diterima, *chatbot* akan mengelompokkan pertanyaan tersebut ke dalam *category* yang sesuai. Kategori ini berfungsi sebagai pengelompokan dasar untuk memetakan berbagai jenis pertanyaan atau permasalahan yang mungkin diajukan oleh mahasiswa.

C. Perancangan Sistem

Rancangan arsitektur sistem menggambarkan cara kerja keseluruhan sistem dengan jelas. Setiap komponen yang diimplementasikan, mulai dari teknologi *cloud* hingga klien, saling terhubung untuk memastikan sistem berfungsi dengan baik. Gambar 3 menggambarkan secara menyeluruh komponen-komponen utama yang membentuk sistem ini dan bagaimana lingkungan sistem dapat beroperasi.

```

<aiml version="2.0">
  <category>
    <pattern>KAMU SIAPA</pattern>
    <template>
      <![CDATA[
      Kamu adalah sebuah AI Layanan Jurusan TIK yang sedang dikembangkan untuk dapat memudahkan mahasiswa jurusan untuk dalam mengakses informasi.
      ]]>
    </template>
  </category>
  <category>
    <pattern>RUANG DOSEN</pattern>
    <template>
      <![CDATA[
      Jurusan TIK mempunyai total 2 ruangan dosen yang terletak sebagai berikut:
      <br/>
      1. Lantai 2 sayap kanan. Ujung koridor ruang 200.
      <br/>
      2. Lantai 2 sayap kiri. Ruang pertama sebelah kanan ruang 200.
      <br/>
      Kamu juga bisa menanyakan lokasi ruang belajar TIK atau laboratorium jika Anda memerlukan informasi lebih lanjut.
      ]]>
    </template>
  </category>
  <category>
    <pattern>KETUA JURUSAN</pattern>
    <template>
      <![CDATA[
      Cari informasi terkait tentang Bapak Muhammad Nasir dengan beberapa keyword dibawah ini:
      <br/>
      1. NIP Bapak Nasir
      <br/>
      2. Gelar Bapak Nasir
      <br/>
      Terimakasih.
      ]]>
    </template>
  </category>
  <category>
    <pattern>APA GELAR PAK SAFRIADI</pattern>
    <template>
      <![CDATA[
      Safriadi, S.T., M. Kom.
      ]]>
    </template>
  </category>
  <category>
    <pattern>RUANGAN BUK MURSYIDAH</pattern>
    <template>
      <![CDATA[
      Ruangan Bu Mursyidah berada di lantai 2 sayap kiri dengan nomor ruangan 200.
      ]]>
    </template>
  </category>
</aiml>
    
```

Gambar 3. Rancangan Sistem

D. Pengujian

Penelitian ini menggunakan metode *Quality of Service* (QoS) untuk menilai efisiensi jaringan dan kinerja *database* aplikasi *chatbot*. Evaluasi dilakukan melalui parameter seperti *throughput*, *delay*, *jitter*, dan *packet loss* untuk mengukur performa jaringan. *Throughput* menilai aliran data yang stabil, sementara *delay* mengukur waktu respons aplikasi. *Jitter* mengamati variasi waktu penerimaan paket, dan *packet loss* mengidentifikasi hilangnya data selama transmisi. Pengukuran ini membantu mengidentifikasi potensi hambatan dan memastikan aplikasi tetap berjalan optimal dalam berbagai kondisi.

III. HASIL DAN PEMBAHASAN

A. Persiapan Data

Tahap persiapan data dimulai dengan pengumpulan dataset. *Dataset* ini terdiri dari 805 data yang telah dikumpulkan untuk analisis. *Dataset* ini terdiri dari 805 data yang telah dikumpulkan untuk analisis.

TABEL 1
SAMPLE DATASET AIML

Pattern	Template
Kamu siapa?	Aku adalah sebuah AI Layanan Jurusan TIK yang sedang dikembangkan untuk dapat memudahkan mahasiswa jurusan untuk dalam mengakses informasi.
Ruang Dosen	Jurusan TIK mempunyai total 2 ruangan dosen yang terletak sebagai berikut: <ol style="list-style-type: none"> Lantai 2 sayap kanan. Ujung koridor ruang 200. Lantai 2 sayap kiri. Ruang pertama sebelah kanan ruang 200. Kamu juga bisa menanyakan lokasi ruang belajar TIK atau laboratorium jika Anda memerlukan informasi lebih lanjut.
Ketua Jurusan TIK	Bapak Ketua Jurusan kita saat ini adalah Bapak Muhammad Nasir, S.T., M.T.
Ketua Jurusan	Beliau mulai menjabat pada Tahun 2024 hingga saat ini. Cari informasi terkait tentang Bapak Muhammad Nasir dengan beberapa keyword dibawah ini: <ol style="list-style-type: none"> NIP Bapak Nasir Gelar Bapak Nasir Terimakasih.
NIP Bapak Atthariq	NIP. 1978072002001121001
Apa Gelar Pak Safriadi?	Safriadi, S.T., M. Kom.
Ruangan Buk Mursyidah	Ruangan Bu Mursyidah berada dilantai 2 sayap kiri dengan nomor ruangan 200.

B. Implementasi *Dataset* pada AIML

Dataset yang sudah ada sebelumnya kemudian di implementasi dalam kodingan berdasarkan metode *Artificial Intelligence Markup Language* untuk membangun *chatbot* berbasis teks. Setiap kategori dalam kode terdiri dari *pattern* yang mendefinisikan input pengguna dan *template* yang berisi respons sistem. Kodingan ini memungkinkan *chatbot* menjawab pertanyaan terkait informasi jurusan Teknologi Informasi dan Komputer, seperti lokasi ruang dosen, data ketua jurusan, dan informasi lain yang dibutuhkan mahasiswa.

```

<br/>
Terimakasih.
</template>
</category>
<category>
<pattern>APA GELAR PAK SAFRIADI</pattern>
<template>
Safriadi, S.T., M. Kom.
</template>
</category>
<category>
<pattern>RUANGAN BUK MURSYIDAH</pattern>
<template>
Ruangan Bu Mursyidah berada di lantai 2 sayap kiri
dengan nomor ruangan 200.
</template>
</category>
</aiml>

```

Gambar 4. Implementasi Dataset Pada Motode AIML

C. Pengujian Metode AIML

Pengujian AIML ini untuk memproses input pengguna dan mencocokkannya dengan pola (pattern) yang telah didefinisikan dalam *dataset* AIML. Sistem ini dirancang agar dapat memberikan tanggapan otomatis berdasarkan *template* jawaban yang sesuai dengan pola pertanyaan pengguna. Jika pola mengandung *wildcard*, kode akan memprosesnya menggunakan ekspresi reguler untuk mendeteksi kecocokan yang lebih fleksibel. Selain itu, elemen random juga ditangani untuk memungkinkan variasi respons acak ketika ada beberapa kemungkinan jawaban.

```

def get_aiml_response(user_input):
    for category in root.findall("./category"):
        pattern = category.find('pattern').text
        # Jika pola cocok secara langsung
        if user_input.lower() == pattern.lower():
            template = category.find('template')
            random_element = template.find('random')
            if random_element is not None:
                return
            return
        process_random_element(random_element)
        else:
            return template.text.strip()
        # Jika pola mengandung wildcard
        elif '*' in pattern:
            pattern_parts = pattern.split('*')
            if pattern_parts[0].lower() in
            user_input.lower() and pattern_parts[1].lower() in
            user_input.lower():
                template = category.find('template')
                random_element = template.find('random')
                if random_element is not None:
                    return
            process_random_element(random_element)
        else:
            return template.text.strip()
    return "Saya tidak mengerti pertanyaan Anda."

```

Gambar 5. Pengujian Metode AIML

Gambar 3 menjelaskan implementasi fungsi `get_aiml_response` yang digunakan untuk menyediakan respons berdasarkan *Artificial Intelligence Markup Language* (AIML) pada sistem *chatbot*. Fungsi ini bekerja dengan mencocokkan pola untuk menentukan respons terhadap input pengguna. Pertama, fungsi membaca struktur

data XML yang berisi kategori AIML. Dari setiap kategori, pola pertanyaan diekstrak dari elemen 'pattern'. Jika pola cocok secara langsung dengan input pengguna (dengan memperhatikan huruf besar/kecil), fungsi mengambil respons dari elemen 'template'. Jika template berisi elemen 'random', fungsi menggunakan `process_random_element` untuk menghasilkan respons acak. Jika tidak, teks dari template yang telah di-strip akan dikembalikan. Untuk pola dengan wildcard '*', fungsi memeriksa kecocokan pola dengan *input* pengguna, lalu mengembalikan *respons* yang sesuai. Jika tidak ada pola yang cocok, fungsi mengirim pesan default "Saya tidak mengerti pertanyaan Anda." Hal ini memungkinkan *chatbot* merespons beragam pertanyaan berdasarkan aturan AIML yang telah ditetapkan..

D. Pengujian Quality Of Service

Pengujian *Quality of Service* (QoS) dilakukan untuk menilai performa layanan *cloud computing* pada aplikasi *chatbot* yang berfungsi sebagai asisten virtual. Pengujian ini menggunakan perangkat lunak *Apache JMeter* untuk mensimulasikan beban *request* dan *Wireshark* untuk menangkap serta merekam data. Parameter yang diukur mencakup *throughput*, *packet loss*, *delay*, dan *jitter*, sesuai dengan standar TIPHON. Pengujian ini akan difokuskan pada *endpoint* API yang bertugas untuk memvalidasi *category* dan *pattern* berdasarkan pesan dari mahasiswa, yaitu *endpoint* `http://54.91.244.200:5000/process_aiml`. Ada beberapa tahap pengujian dengan jumlah pengguna virtual atau *thread* yang bervariasi, dengan jumlah maksimal antara 50 hingga 900 berdasarkan jumlah mahasiswa jurusan, dan *ramp up period* yang sama yaitu selama satu detik untuk semua percobaannya. Berikut ini adalah analisis mendalam terhadap hasil pengujian berdasarkan tabel parameter yang diperoleh.

TABEL 4
HASIL PENGUJIAN THROUGHPUT QoS

Virtual User	Throughput	Indeks	Kategori
50	225 Kbps	0	Buruk
100	447 Kbps	1	Kurang Baik
200	860 Kbps	2	Cukup
400	1334 Kbps	3	Baik
600	1783 Kbps	3	Baik
800	2620 Kbps	4	Sangat Baik
900	4108 Kbps	4	Sangat Baik

Berdasarkan Tabel 4, hasil pengujian *throughput* menunjukkan variasi performa dengan peningkatan jumlah *virtual user*. Pada 50 *virtual user*, *throughput* menunjukkan 225 Kbps dengan kategori buruk. Seiring meningkatnya jumlah pengguna, *throughput* meningkat signifikan hingga mencapai 4108 Kbps pada 900 *virtual user* dengan kategori sangat baik. Kenaikan *throughput* yang konsisten menunjukkan bahwa sistem dapat mengalokasikan sumber daya secara efisien sesuai dengan beban yang diberikan. Hal ini menunjukkan bahwa sistem mampu menangani beban

tinggi dengan efisiensi tinggi, namun memiliki masalah pada beban rendah.

TABEL 5
HASIL PENGUJIAN *PACKET LOSS* QoS

Virtual User	Packet Loss	Indeks	Kategori
50	0 %	4	Sangat Baik
100	0.1 %	4	Sangat Baik
200	0.2 %	4	Sangat Baik
400	0.2 %	4	Sangat Baik
600	0.5 %	4	Sangat Baik
800	0.4 %	4	Sangat Baik
900	0.6 %	4	Sangat Baik

Hasil pengujian *packet loss* menunjukkan performa yang sangat baik hingga baik. Dari Tabel 5, hasil pengujian menunjukkan *packet loss* yang sangat minimal, dengan nilai tertinggi hanya 0.6% pada 900 *virtual user*. Semua kategori berada pada indeks sangat baik, yang menunjukkan jaringan stabil dengan kehilangan paket yang sangat rendah, sehingga data dapat dikirimkan dengan efisien.

TABEL 6 HASIL PENGUJIAN *DELAY* QoS

Virtual User	Delay	Indeks	Kategori
50	0.3 ms	4	Sangat Baik
100	0.1 ms	4	Sangat Baik
200	0 ms	4	Sangat Baik
400	0 ms	4	Sangat Baik
600	0 ms	4	Sangat Baik
800	0 ms	4	Sangat Baik
900	0 ms	4	Sangat Baik

Berdasarkan Tabel 6, hasil pengujian *delay* menunjukkan performa yang sangat baik di semua level *virtual user*. Performa *delay* yang sangat baik pada semua tingkat beban menunjukkan sistem yang responsif dan efisien dalam penanganan permintaan data.

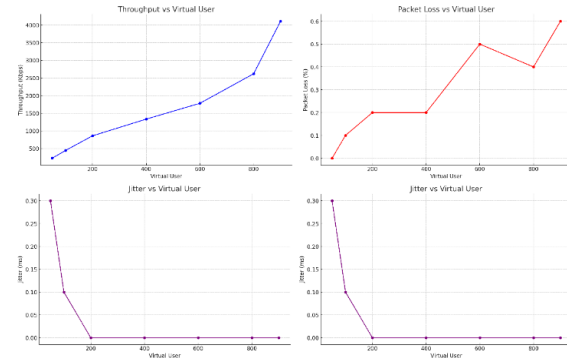
TABEL 7 HASIL PENGUJIAN *JITTER* QoS

Virtual User	Jitter	Indeks	Kategori
50	0.3 ms	4	Sangat Baik
100	0.1 ms	4	Sangat Baik
200	0 ms	4	Sangat Baik
400	0 ms	4	Sangat Baik
600	0 ms	4	Sangat Baik
800	0 ms	4	Sangat Baik
900	0 ms	4	Sangat Baik

Berdasarkan Tabel 7, hasil pengujian *jitter* menunjukkan variasi performa yang sangat baik. Hasil pengujian *jitter* menunjukkan nilai yang sangat rendah, mencapai 0 ms pada 200 *virtual user* hingga 900 *virtual user*. Ini menunjukkan kestabilan waktu pengiriman paket data.

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem mampu menangani peningkatan beban dengan sangat baik, meningkatkan efisiensi dan kualitas layanan jaringan secara keseluruhan. Evaluasi parameter QoS ini memberikan pemahaman mendalam tentang efektivitas layanan jaringan dan performa *database*, yang penting untuk memastikan aplikasi berfungsi optimal dalam kondisi beban

yang beragam. Perbandingan nilai parameter pada berbagai tingkat beban dapat dianalisis melalui grafik yang disajikan pada Gambar 4.



Gambar 4. Grafik Parameter Quality Of Service

IV. KESIMPULAN

Evaluasi terhadap parameter *Quality of Service* (QoS) menunjukkan bahwa sistem memiliki performa yang sangat baik dalam hal *packet loss*, *delay*, dan *jitter* pada beban tinggi, dengan *throughput* yang relatif stabil. Meskipun demikian, terdapat penurunan performa *throughput* pada beban rendah. Analisis ini menekankan pentingnya peningkatan kualitas jaringan agar layanan tetap optimal dan andal di berbagai kondisi beban sistem..

REFERENSI

- [1] Guntoro, Costaner, L., & Lisnawati. (2020). Aplikasi Chatbot Untuk Layanan Informasi Dan Akademik Kampus Berbasis Artificial Intelligence Markup Language (Aiml). *Jurnal Teknologi Informasi & Komunikasi*, 11, 291–300. <https://doi.org/10.31849/Digitalzone.V11i2.5049iccs>
- [2] Amrullah, A. Z., Anas, A. S., & Primajati, G. (2022). Implementasi Chatbot Sebagai Virtual Assistant Penerimaan Mahasiswa Baru Pada Universitas Bumigora. *Bumigora Information Technology (Bite)*, 4(1), 17–26. <https://doi.org/10.30812/Bite.V4i1.1664>
- [3] Muhammad, F. Z., & Hermawan, S. (2023). Pemanfaatan Artificial Intelligence Markup Language (Aiml) Dan Latent Semantic Analysis (Lsa) Dalam Pengembangan Chatbot E-Education Utilization Of Artificial Intelligence Markup Language (Aiml) And Latent Semantic Analysis (Lsa) In E-Education Chatbot Development. *Journal Of Information Technology And Computer Science (IntecomS)*, 6(1), 36–43.
- [4] Huwaida, H., Rofi'i, & Imelda, S. (2018). Pengaruh Kualitas Pelayanan Pendidikan Terhadap Kepuasan Mahasiswa Di Jurusan Administrasi Bisnis Politeknik Negeri Banjarmasin. *Intekna*, 18, 76–131.
- [5] Supangat, Mohd, Z. B. S., & Roenadi, K. (2021). Pemanfaatan Artificial Intelligence (Ai) Sebagai Respon Learning Style Mahasiswa. *Konferensi Nasional Ilmu Komputer (Konik)*, 270–279.
- [6] Putra, D. H., Asfi, M., & Fahrudin, R. (2021). Perancangan Ui/Ux Menggunakan Metode Design Thinking Berbasis Web Pada Laportea Company. *Dalam Rifqi Fahrudin Jurnal Ilmiah Teknologi Informasi Terapan (Vol. 8, Nomor 1)*.
- [7] Setiawan, W., Fajriyah, N., & Duha, T. (2022). Analisa Layanan Cloud Computing Di Era Digital. *Jurnal Informatika*, 1, 32–39.
- [8] Kusnanda, I. G. R., Sukarsa, I. M., & Susila, A. A. N. H. (2022). Perancangan Chatbot Hotel Dengan Model Natural Language Processing Chatbot Dan Button Based Chatbot.
- [9] Adi, N. H., Basar, A. R., Umar, S., & Trilidia Devega, A. (2022). Virtual Asisten Pusat Informasi Mahasiswa Berbasis Android Menggunakan Flutter. *Rintvet | Research In Technical And Vocational*

- Education And Training , 1(2), 79–90.
<https://doi.org/10.55585/Rintvet.V1i2.16>
- [10] Aruriansyah, S. N., Cherid, A., Santoso, H., & Rochmah, D. A. (2023). Rancang Bangun Lingkungan Pemrograman Python Dengan Metode Chatbot Pada Platform Whatsapp (Vol. 20, Nomor 2).
- [11] Mubarak, A. (2019). Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek. Dalam Jurnal Informatika Dan Komputer Ternate (Vol. 02, Nomor 1).
- [12] Uminingsih, Muhammad, N. I., Muhaamad, Y., & Suraya. (2022). Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula Info Artikel Abstrak. Jurnal Ilmiah Teknik Dan Ilmu Komputer, 1(2), 1–8.
<https://doi.org/10.55123>
- [13] Nisa, I. S., Saputro, R. M., Nugroho, T. F., & Lahitani, A. R. (2024). Analisis Quality Of Service (Qos) Menggunakan Standar Parameter Tiphon Pada Jaringan Internet Berbasis Wi-Fi Kampus 1 Unjaya. Teknomatika: Jurnal Informatika Dan Komputer, 17(1), 1–9.
<https://doi.org/10.30989/Teknomatika.V17i1.1307>