

# Perancangan Chatbot FAQ Dengan Algoritma Naive Bayes Classifier Untuk Layanan Umum

<sup>1\*</sup>Fahrezian Arya Bima, <sup>2</sup>Ardi Sanjaya, <sup>3</sup>Risky Aswi Ramadhani

<sup>1, 2, 3</sup> Teknik Informatika, Universitas Nusantara PGRI Kediri

E-mail: <sup>1</sup>[faaryabima@gmail.com](mailto:faaryabima@gmail.com), <sup>2</sup>[dersky@gmail.com](mailto:dersky@gmail.com), <sup>3</sup>[riskyaswiramadhani@gmail.com](mailto:riskyaswiramadhani@gmail.com)

*Penulis Korespondens : Ardi Sanjaya*

**Abstrak**— *Chatbot* adalah sebuah program yang dirancang untuk melakukan simulasi percakapan manusia. Penelitian ini menggunakan chatbot sebagai solusi untuk menggantikan layanan pengguna masyarakat dengan otomatis yang mampu menjawab pertanyaan umum seputar layanan umum masyarakat. Pengumpulan dataset tentang pertanyaan yang sering ditanyakan oleh masyarakat dilakukan dengan cara wawancara terhadap petugas layanan umum masyarakat. Model algoritma Naïve Bayes Classifier digunakan dalam pemodelan sistem chatbot. Hasil evaluasi yang dilakukan menghasilkan nilai akurasi sebesar 69.23%, presisi 58.97%, recall 69.23%, dan f1-score 60.51%. *Chatbot* berhasil menjawab 8 pertanyaan yang sesuai dari 10 pertanyaan yang dilakukan. Penelitian ini berhasil mengintegrasikan model *chatbot* kedalam basis web menggunakan Django sebagai server simulasinya. Model chatbot perlu dilakukan peninjauan kembali agar performa yang dihasilkan bisa lebih baik lagi.

**Kata Kunci**— Chatbot, FAQ, Naïve Bayes, Web

**Abstract**— *Chatbot* is a program designed to simulate human conversation. This study uses chatbot as a solution to replace public user services with automatic ones that are able to answer general questions about public services. The collection of datasets about questions frequently asked by the public was carried out by interviewing public service officers. The Naïve Bayes Classifier algorithm model was used in modeling the chatbot system. The evaluation results produced an accuracy value of 69.23%, precision 58.97%, recall 69.23%, and f1-score 60.51%. The chatbot managed to answer 8 appropriate questions out of 10 questions asked. This study successfully integrated the chatbot model into a web base using Django as its simulation server. The chatbot model needs to be reviewed so that the resulting performance can be even better.

**Keywords**— Chatbot, FAQ, Naïve Bayes, Web

This is an open access article under the CC BY-SA License.



## I. PENDAHULUAN

Di lingkungan pemerintahan, penyampaian informasi yang cepat dan layanan yang responsif menjadi salah satu permasalahan yang krusial dalam meningkatkan kecepatan proses administrasi dan juga kepuasan masyarakat. *Chatbot* adalah sebuah program yang dirancang untuk melakukan simulasi percakapan manusia. Pada konteks ini, *chatbot* dapat berfungsi sebagai layanan pengguna otomatis yang mampu menjawab pertanyaan umum seputar layanan umum masyarakat. Penelitian yang dilakukan Simon Pranata Baurus dan Evalien Surijati pada tahun 2021, berhasil mengembangkan sebuah *chatbot* dengan menggunakan Dialogflow untuk

melayani *FAQ* di perpustakaan Universitas Matana, sedangkan pada penelitian ini berfokus pada perancangan *chatbot FAQ* untuk layanan umum/masyarakat[1].

*Chatbot* merupakan suatu program komputer yang memiliki kecerdasan buatan yang berbentuk simulasi percakapan interaktif antara mesin dan manusia melalui teks, suara, dan visual atau gambar[2]. *Chatbot FAQ* merupakan sistem yang dirancang untuk menjawab pertanyaan umum yang ada dalam pikiran orang-orang dan juga yang sering ditanyakan oleh orang-orang[3]. *Chatbot FAQ* mengenali masukan pengguna dengan cara mencocokkan pola dengan database *FAQ*[3]. *Chatbot* sangat bergantung pada data pengetahuan yang sudah dibuat sebelumnya dari pengembang sistem[2]. *Chatbot* biasanya digunakan untuk kegiatan layanan konsumen, yang dimana *chatbot* berperan sebagai customer service yang akan secara otomatis memberikan *feedback* yang sesuai dengan apa yang pengguna berikan[4].

Permasalahan yang terkait dengan banyaknya pertanyaan yang sering berulang dari masyarakat terhadap layanan umum seperti layanan pemerintahan seringkali mengakibatkan beban kerja tambahan bagi para petugas. Dengan ini, diperlukan sebuah sistem yang mampu memberikan sebuah jawaban kepada masyarakat atau pengguna secara otomatis dan juga konsisten. Maka dari itu, penelitian ini bertujuan untuk membangun sebuah sistem *chatbot* Frequently Asked Questions (FAQ) yang dapat membantu menjawab pertanyaan-pertanyaan umum pengguna terkait layanan umum masyarakat secara otomatis dan juga cepat.

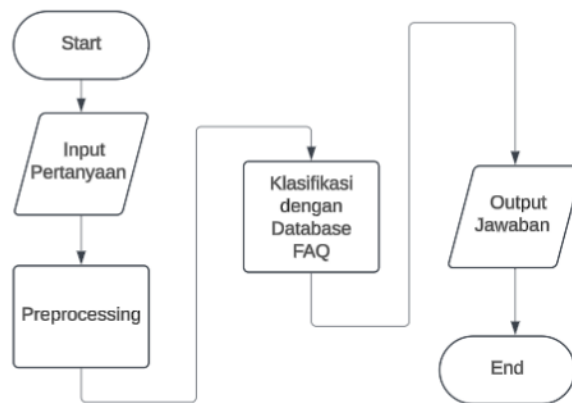
## II. METODE

### A. Tahapan Penelitian

Pada penelitian ini penulis melakukan beberapa tahap, yaitu melakukan pengumpulan beberapa studi literatur untuk dijadikan referensi yang berkaitan dengan penelitian tentang pembuatan sistem *Chatbot*. Tahap selanjutnya melakukan pengumpulan dataset yang dibutuhkan dengan teknik wawancara terhadap petugas layanan umum masyarakat. Data yang berhasil dikumpulkan sebelumnya diolah menjadi sumber informasi dengan demikian karakteristik agar data tersebut dapat dipahami dan bisa bermanfaat sebagai solusi dari permasalahan pada penelitian ini [5]. Lalu dilakukan pembangunan model menggunakan dataset yang sudah dikumpulkan menggunakan algoritma Naïve Bayes Classifier. Dan yang terakhir pembangunan user interface berbasis web menggunakan Django sebagai server simulasinya.

### B. Desain Sistem

Tahap desain merupakan rincian dari perencanaan sistem sesuai spesifikasi yang telah ditetapkan. Yang mencakup beberapa struktur dari arsitektur, alur kerja, rancangan komponen, dan perencanaan teknis implementasi solusi yang dikembangkan [6].



Gambar 1. Desain sistem *Chatbot*

Pada gambar 1 menampilkan sebuah alur dari sistem *chatbot*. Sistem *chatbot* yang akan dirancang ini memiliki empat tahap utama yaitu, input pertanyaan dari user, lalu input pertanyaan dari pengguna dilakukan preprocessing terlebih dahulu seperti penghapusan tanda baca, merubah huruf/kalimat ke bentuk low sentence case, tokenisasi, dan yang terakhir stemming, setelah itu input pertanyaan user yang telah dilakukan preprocessing dilakukan proses klasifikasi menggunakan algoritma Naïve Bayes Classifier, dan terakhir menghasilkan output jawaban kepada user.

#### C. Dataset

Dataset yang digunakan didapat dari kantor kelurahan peneliti, dataset disusun secara manual berdasarkan kumpulan pertanyaan-pertanyaan yang sering ditanyakan oleh masyarakat sekitar. Dataset terdiri dari 30 pertanyaan yang dimasukkan kedalam beberapa kategori label (jawaban). Setiap data terdiri atas pertanyaan dan label (jawaban). Contoh data yang digunakan ditunjukkan pada Tabel 1.

Tabel 1. Dataset *FAQ*

<i>No</i>	<i>Pertanyaan</i>	<i>Jawaban</i>
1	Apa syarat membuat SKTM?	Persyaratan dalam membuat SKTM adalah ...
2	Apa syarat membuat Kartu Keluarga?	Persyaratan dalam membuat Kartu Keluarga adalah...
3	Apa syarat membuat Akta Kelahiran?	Persyaratan dalam membuat Akta Kelahiran adalah ...

#### D. Preprocessing

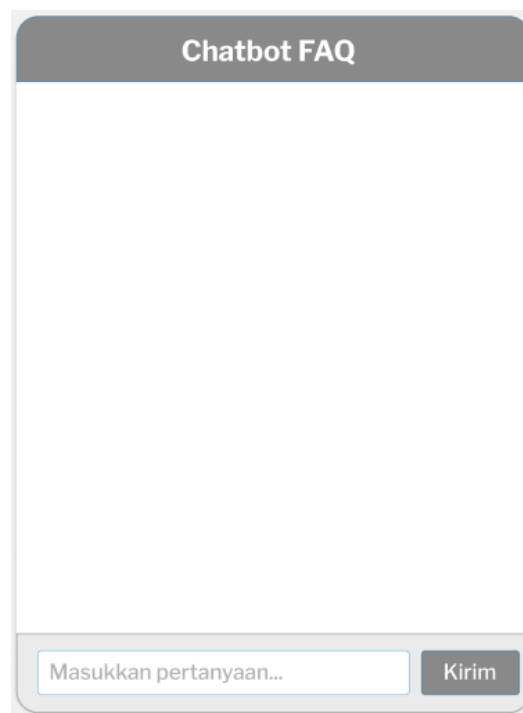
Sebelum dilakukan proses klasifikasi dengan algoritma Naïve Baiyes, data teks akan dilakukan tahap preprocessing terlebih dahulu untuk meningkatkan kualitas data. Preprocessing merupakan serangkaian langkah yang dilakukan untuk menyiapkan data yang sebelumnya mentah agar siap digunakan dalam proses analisis dalam *machine learning* [7]. Preprocessing dapat memberikan peningkatan dengan menerapkan langkah-langkah preprocessing yang tepat, yang dapat mengatur kalimat dengan cara yang lebih baik dan membantu dalam menghasilkan respon yang sesuai [8]. Tahapan preprocessing yang dilakukan sistem ini adalah penghapusan

tanda baca, perubahan bentuk huruf yang sebelumnya Sentence case dirubah menjadi lowercase, tokenisasi yang berfungsi untuk memecah kalimat menjadi kata-kata, dan yang terakhir stemming yang berfungsi untuk mengembalikan kata yang sebelumnya sudah ditokenisasi menjadi bentuk kata dasarnya. Pada tahapan ini library yang digunakan adalah library NLTK dan library Sastrawi.Stemmer.

#### E. Perancangan Model Algoritma

Model algoritma yang akan digunakan untuk proses klasifikasi adalah Naïve Bayes. Naïve Bayes Classifier adalah algoritma klasifikasi yang berbasis probabilistik, yang dimana menerapkan teorema bayes dengan asumsi kuat bahwa setiap fitur bersifat independen satu sama lain [9]. Algoritma Naïve Bayes bekerja dengan mengklasifikasikan teks kedalam kategori tertentu agar *chatbot* dapat mengidentifikasi maksud dari pengguna, karena identifikasi maksud dari pengguna merupakan salah satu langkah awal dan terpenting dari percakapan *chatbot* [10]. Menggunakan Pipeline dari scikit-learn untuk menyederhanakan proses preprocessing dan feature extraction (CountVectorizer dan TfidfTransformer) dengan model klasifikasi (MultinomialNB) dalam satu alur yang konsisten dan juga efisien.

#### F. Desain UI



Gambar 2. Desain UI

Desain dari tampilan UI ditunjukkan pada Gambar 2.

#### G. Evaluasi

Pada tahap evaluasi dilakukan beberapa proses untuk mengetahui hasil akhir dari performa model algoritma Naïve Bayes Classifier yang telah dikembangkan, Evaluasi pemodelan dilakukan setelah proses pemodelan selesai dilakukan [11], dengan membagi dataset menjadi data training dan data test dengan perbandingan 60:40. Evaluasi dilakukan dengan

menggunakan beberapa metrik pengujian untuk mengetahui nilai akurasi, presisi, recall, f1-score, confusion matrix dan pengujian fungsionalitas *chatbot* menggunakan *Black Box Testing*.

#### H. Integrasi

Integrasi dari model *chatbot* diimplementasikan kedalam web menggunakan Django sebagai server simulasi dan dijalankan secara lokal. Model algoritma Naïve Bayes Classifier yang telah dikembangkan sebelumnya disimpan menjadi file berformat pickle, yang lalu akan dijalankan oleh server Django sebagai backend atau knowledge base dari web *chatbot*.

### III. HASIL DAN PEMBAHASAN

Pengujian dilakukan dengan cara mengambil salah satu sample data dari 30 total data yang digunakan untuk menguji kinerja dari *chatbot* dan menjelaskan cara kerja *chatbot* dari tahap preprocessing hingga tahap klasifikasi. Contoh sampel data yang digunakan ditunjukkan pada tabel 2.

Tabel 2. Data sampel Preprocessing

No	Pertanyaan	Jawaban
1	Apa syarat membuat SKTM?	Persyaratan dalam membuat SKTM adalah ...

Tahap pertama pertanyaan akan dilakukan proses preprocessing untuk agar teks dari inputan dapat dikenali dan juga dapat diolah oleh algoritma Naïve Bayes Classifier. Yang pertama yaitu tahap penghapusan tanda baca, hasil penghapusan tanda baca dapat ditunjukkan pada gambar 3.

Nopunch : Apa syarat membuat SKTM

Gambar 3. Hasil penghapusan tanda baca

Setelah dilakukan preprocessing berupa penghapusan tanda baca, pertanyaan akan dilakukan tahap preprocessing selanjutnya yaitu merubah bentuk huruf yang sebelumnya Sentence case menjadi lower case, hasil preprocessing dapat ditunjukkan pada gambar 4.

lowercase : apa syarat membuat sktm

Gambar 4. Hasil lowercase

Tahap selanjutnya setelah dilakukan tahap lowercase, pertanyaan akan dilakukan preprocessing berupa tokenisasi, hasil preprocessing dapat dilihat pada gambar 5.

Tokenisasi : ['apa', 'syarat', 'membuat', 'sktm']

Gambar 5. Hasil Tokenisasi

Setelah dilakukan tahap preprocessing berupa tokenisasi, selanjutnya tahap terakhir dalam proses preprocessing yaitu stemming, hasil stemming dapat dilihat pada gambar 6.

Stemming : ['apa', 'syarat', 'buat', 'sktm']

Gambar 6. Hasil Stemming

Setelah tahap preprocessing selesai, tahap selanjutnya sebelum masuk kedalam proses klasifikasi yaitu tahap *feature extraction* yang memiliki dua proses yaitu *CountVectorizer* dan *TfidfTransformer*. Pengujian dilakukan dengan mengambil dua sample data sebagai pembandingan pada tahap *feature extraction*. Contoh sample data yang digunakan ditunjukkan pada tabel 3.

Tabel 3. Data sampel *Feature extraction*

No	Pertanyaan	Jawaban
1	Apa syarat membuat SKTM?	Persyaratan dalam membuat SKTM adalah ...
2	Apa syarat membuat Kartu Keluarga?	Persyaratan dalam membuat Kartu Keluarga adalah ...

Tahap pertama dari *feature extraction* adalah *CountVectorizer*, *CountVectorizer* berfungsi untuk mengubah teks pertanyaan menjadi representasi vektor berdasarkan jumlah kemunculan kata atau *bag-of-words*. Hasil *CountVectorizer* dapat ditunjukkan pada tabel 4.

Tabel 4. Hasil *CountVectorizer*

No	Pertanyaan	Nilai <i>CountVectorizer</i>
1	Apa syarat membuat SKTM?	[1, 1, 0, 0, 1, 1]
2	Apa syarat membuat Kartu Keluarga?	[1, 1, 1, 1, 0, 1]

Tahap selanjutnya dari *feature extraction* adalah *TfidfTransformer*. *TfidfTransformer* berfungsi untuk mengubah nilai hasil dari *CountVectorizer* menjadi nilai TF-IDF atau pembobotan, yang dimana semakin jarang atau langka kata muncul, maka bobot semakin besar atau kata semakin penting. Hasil dari *TfidfTransformer* ditunjukkan pada tabel 5.

Tabel 5. Hasil *TfidfTransformer*

No	Kata	DF	IDF
1	sktm	1	1.405
2	kartu	1	1.405
3	keluarga	1	1.405
4	apa	2	1.0
5	syarat	2	1.0
6	buat	2	1.0

Pada tabel 5, bisa dilihat bahwa kata yang sering muncul dalam dataset atau pertanyaan akan memiliki nilai pembobotan yang lebih kecil atau bisa disebut dengan kata yang kurang penting. Sebaliknya, kata yang jarang muncul dalam pertanyaan akan memiliki bobot yang lebih tinggi atau dianggap penting, sehingga lebih berpengaruh dalam proses klasifikasi. Dengan ini, proses *feature extraction* bisa dikatakan selesai dan selanjutnya akan dilakukan proses klasifikasi dengan algoritma Naïve Bayes Classifier.

Selanjutnya dilakukan pengujian Black Box Testing, pengujian ini berfokus pada fungsionalitas dari sistem *chatbot* yang telah dikembangkan. Pengujian *Black Box Testing* bertujuan untuk menguji kemampuan dari sistem *chatbot* dalam menghasilkan respon yang

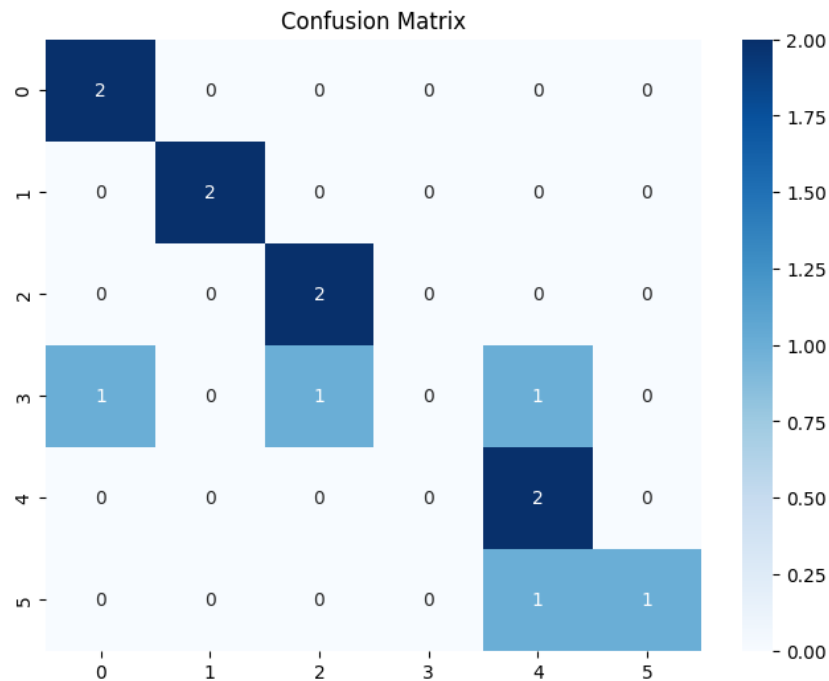
sesuai atau tidak. Hasil pengujian *chatbot* menggunakan *Black Box Testing* ditunjukkan pada tabel 6.

Tabel 6. *Black Box Testing chatbot*

No	Pertanyaan	Response / Jawaban	Hasil Sesuai / Tidak Sesuai
1	Apa syarat membuat SKTM?	Persyaratan dalam membuat SKTM adalah ...	Sesuai
2	Apa syarat membuat Kartu Keluarga?	Persyaratan dalam membuat Kartu Keluarga adalah...	Sesuai
3	Apa syarat membuat Akta Kelahiran?	Persyaratan dalam membuat Akta Kelahiran adalah ...	Sesuai
4	Bagaimana cara membuat surat Akta Kelahiran?	Persyaratan dalam membuat Akta Kelahiran adalah ...	Sesuai
5	Bagaimana cara membuat surat SKCK?	Persyaratan dalam membuat SKCK adalah ...	Tidak Sesuai
6	Apa saja yang dibutuhkan untuk membuat surat SKCK?	Persyaratan dalam membuat SKCK adalah ...	Tidak Sesuai
7	Apa syarat membuat surat keterangan pertanian?	Persyaratan dalam membuat surat keterangan pertanian adalah ...	Sesuai
8	Bagaimana cara membuat surat kematian?	Persyaratan dalam membuat surat kematian adalah ...	Sesuai
9	Apa saja yang dibutuhkan untuk membuat surat kematian?	Persyaratan dalam membuat surat kematian adalah ...	Sesuai
10	Apa saja dokumen yang diperlukan untuk membuat surat keterangan kematian?	Persyaratan dalam membuat surat keterangan kematian adalah ...	Sesuai

Berdasarkan hasil pengujian menggunakan *Black Box Testing*, sistem *chatbot* berhasil menjawab 8 pertanyaan dengan sesuai. Bisa disimpulkan bahwa sistem *chatbot* menunjukkan kemampuan yang cukup baik dalam menghasilkan respon yang sesuai, perlu dilakukan peninjauan kembali agar sistem *chatbot* dapat berjalan lebih baik lagi. Ada beberapa metrik yang digunakan untuk pengujian performa, yaitu confusion matrix, matrik akurasi, presisi, recall, dan f1-score. Menggunakan bantuan library dari sklearn untuk menghitung nilai performa dari model yang dirancang.

Hasil dari perancangan sistem *chatbot* ini, yang melalui beberapa tahapan seperti pengambilan data, *preprocessing*, *feature extraction*, pemodelan algoritma, dan integrasi model mendapatkan hasil nilai evaluasi seperti pada gambar 7 dan 8.



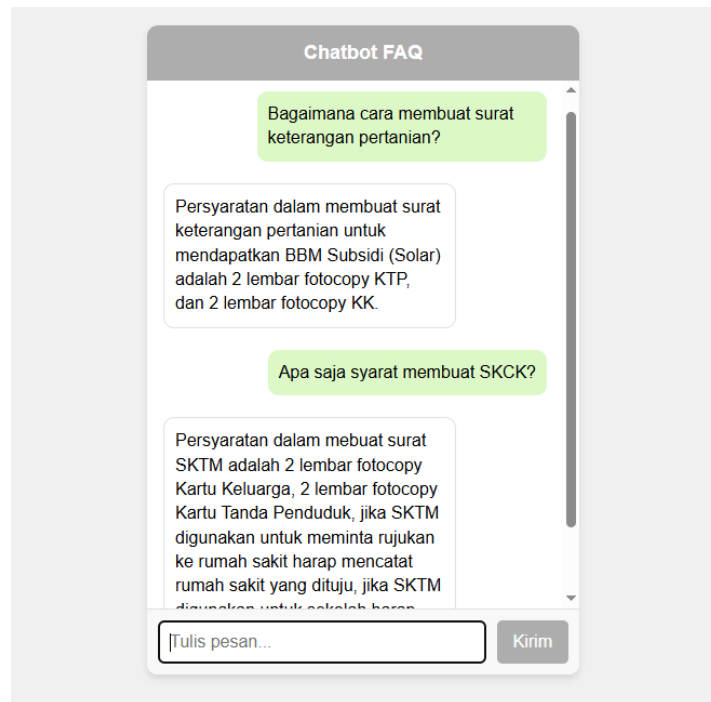
Gambar 7. Hasil Confussion Matrix

Accuracy : 69.23%  
Precision : 58.97%  
recall : 69.23%  
F1 Score : 60.51%

Gambar 8. Hasil evaluasi performa

Evaluasi kinerja model dilakukan dengan menggunakan beberapa matrik pengujian, yaitu confusion matrix, akurasi, presisi, recall, dan F1-score. Berdasarkan hasil pengujian model yang telah dilakukan, pada gambar 8 menunjukkan bahwa model mampu mengklasifikasikan sebagian besar label (jawaban) dengan cukup baik. Pada label 3, model tidak berhasil mengklasifikasikan data dengan sesuai, sementara untuk label 5 hanya setengah data yang diklasifikasikan dengan benar, sisanya diklasifikasikan ke label yang salah. Pada gambar 9, diperoleh nilai akurasi sebesar 69.23%, presisi 58.97%, recall 69.23%, dan f1-score 60.51%, yang dimana model mampu mengklasifikasikan sebagian data dengan benar.



Gambar 97. Tampilan pengguna *chatbot*

Pada gambar 9, dapat disimpulkan bahwa model *chatbot* berhasil diintegrasikan kedalam sistem yang berbasis web, dengan menggunakan server Django sebagai media simulasinya. Integrasi ini membuktikan bahwa chatbot dapat berjalan dengan baik dalam lingkungan web dan juga mampu merespon beberapa pertanyaan yang sesuai dengan data yang telah dilatih menggunakan algoritma Naïve Bayes Classifier.

#### IV. KESIMPULAN

Berdasarkan hasil evaluasi yang dilakukan melalui metrik performa seperti akurasi, presisi, recall, f1-score dan pengujian fungsionalitas menggunakan *Black Box Testing*, chatbot yang dirancang menggunakan algoritma Naïve Bayes Classifier menunjukkan kinerja yang cukup baik. Dengan nilai akurasi sebesar 69.23%, presisi 58.97%, recall 69.23%, dan f1-score 60.51%. sedangkan pengujian fungsionalitas menggunakan *Black Box Testing*, chatbot berhasil menjawab 8 pertanyaan yang sesuai dari 10 pertanyaan yang dilakukan. Model *chatbot* berhasil diintegrasikan kedalam sistem berbasis web menggunakan Django sebagai server simulasinya. Secara keseluruhan, penelitian ini berhasil untuk merancang atau membuat sistem *chatbot* menggunakan algoritma Naïve Bayes Classifier yang diintegrasikan kedalam basis web.

#### DAFTAR PUSTAKA

- [1] S. P. Barus, E. Surijati, "Chatbot with Dialogflow for FAQ Services in Matana University Library". (2021). *International Journal of Informatics and Computations*, 3(2). <https://doi.org/10.35842/ijicom.v3i2.43>.
- [2] F. Mustakim, Fauziah, and N. Hayati, "Algoritma Artificial Neural Network pada Text-based Chatbot Frequently Asked Question (FAQ) Web Kuliah Universitas Nasional".

- (2021). *JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, 5(4). <https://doi.org/10.35870/JTIK.V5I4.261>.
- [3] Farhana Sethi, "FAQ (Frequently Asked Questions) ChatBot for Conversation". (2020). *International Journal of Computer Sciences and Engineering*, 8(10). <https://doi.org/10.22541/au.160435632.23888308/v1>.
- [4] R. Y. Wardani, H. Hermanto, and M. Aly Arosyid, "'Merchan-Chat Bot': Teman Virtual AI Berbasis Android". (2023). *Seminar Nasional Teknologi & Sains*, 2(1). <https://doi.org/10.29407/stains.v2i1.2909>.
- [5] A. Almustaqim, and A. N. Toscani, "Perancangan Sistem Chatbot Sebagai Virtual Assistant Pada PT.Everbright Jambi". (2022). *Sistem Komputer dan Teknik Informatika*, 5(2). <https://doi.org/10.36080/skanika.v5i2.2953>.
- [6] M. A. Nadzif, Saefurrohman, and R. Soelistijadi, "Penggunaan Teknologi Natural Language Processing dalam Sistem Chatbot Untuk Peningkatan Layanan Informasi Administrasi Publik". (2024). *Indonesian Journal of Computer Science*, 13(1). <https://doi.org/10.33022/ijcs.v13i1.3645>.
- [7] S. sutarni, E. Prasetyo, and L. Sudiaty, "Evaluasi Kinerja Chatbot dengan Integrasi Algoritma RNN dan LSTM dalam Optimalisasi respon percakapan pada sistem PMB". (2024). *jsoscied*, 7(2). <https://doi.org/10.33022/ijcs.v13i1.3645>.
- [8] M. Aleedy, and H. Shaiba, "Generating and Analyzing Chatbot Responses using Natural Language Processing". (2019). *International Journal of Advanced Computer Science and Applications*, 10(9). <http://dx.doi.org/10.14569/IJACSA.2019.0100910>.
- [9] K. Vembandasamy, R. Sasipriya, and E. Deepa, "Heart Diseases Detection Using Naïve Bayes Algorithm". (2015). *International Journal of Innovative Science, Engineering & Technology*, 2(9).
- [10] V. Vijayaraghavan, J. B. Cooper, Rian Leevinson J., "Algorithm Inspection for Chatbot Performance Evaluation". (2020). *Procedia Computer Science*, 171. <https://doi.org/10.1016/j.procs.2020.04.245>.
- [11] Sujacka Retno, R. K. Dinata, and N. Hasdyna, "Evaluation of chatbot data model in natural language processing using k-nearest neighbor". (2023). *Jurnal CoSciTech (Computer Science and Information Technology)*, 4(1). <https://doi.org/10.37859/coscitech.v4i1.4690>.