Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 7 No. 1 Januari 2025 Hal. 126-132
http://jurnal.unidha.ac.id/index.php/jteksis

E-ISSN :**2655-8238**
P-ISSN : **2964-2132**

Evaluation of Chatbot Responsiveness and Accuracy Using a Matching Pattern Algorithm for Implementation in a Medium-Sized Enterprise

Haeruddin[a], Sabariman[b], Celvin[c]
[a]Information Technology, Computer Science, Universitas Internasional Batam, haeruddin@uib.ac.id
[b]Information Technology, Computer Science, Universitas Internasional Batam, sabariman@uib.ac.id
[c]Information Technology, Computer Science, Universitas Internasional Batam, 2132042.celvin@uib.edu

Submitted: 11-01-2025, Reviewed: 15-01-2025, Accepted 20-01-2025
https://doi.org/10.47233/jteksis.v5i1.1816

Abstract
PT XYZ is a company specializing in the distribution of high-quality industrial supplies and hardware equipment. One of the key challenges faced by the Store Department is managing requests from other departments, such as inquiries about inventory levels or requests for specific item photos stored in the warehouse. These tasks are often time-consuming due to the large volume of items. To address this issue, this study aims to implement a pattern matching algorithm in a chatbot to support the operations of the Store Department at PT XYZ. The chatbot is designed to perform tasks such as adding data to the database, adjusting warehouse items, displaying item photos, and providing item quantity information. The development process employs the System Development Life Cycle (SDLC) methodology to ensure a systematic and error-minimized approach. The system's performance was analyzed using black box testing to evaluate the functionality of the implemented features. The results demonstrate that the web-based chatbot operates effectively, significantly reducing the workload of users, particularly in inventory management tasks. Through its structured development, the chatbot has successfully enhanced efficiency and provided substantial operational support to the Store Department.
Keywords: *Chatbot, Matching Pattern, Inventory Management, SDLC, Black Box Testing*

INTRODUCTION

AI-based chatbots have become widely utilized in recent years, with applications ranging from assisting in information retrieval, such as ChatGPT, virtual assistants, to customer service. Artificial Intelligence (AI) refers to the technological advancement of computer systems that are designed to mimic the human brain's thinking and operational processes. AI possesses various capabilities, including learning, decision-making, and intelligent reasoning for problem-solving. Through computational analysis, AI systems are developed to perform tasks and make decisions that assist users effectively [1], [2]. The implementation of AI-based chatbots requires specific processes and algorithms to ensure smooth functionality and uninterrupted operation [3]. Several algorithms are commonly applied in chatbot development, including Natural Language Processing (NLP), Deep Learning, Machine Learning, and Pattern Matching, among others, which are frequently used by developer [4], [5].

PT XYZ is a company specializing in the distribution of high-quality industrial supplies and other hardware equipment. One of the key departments at PT XYZ is the Store Department, which is responsible for managing inventory within the warehouse. A significant challenge faced by the Store Department arises when requests from other departments are made, such as inquiries regarding the current inventory levels or requests for photos of specific items within the warehouse. These tasks can be time-consuming due to the large volume of items in the warehouse. Consequently, there is a need for a streamlined management system to expedite the fulfillment of such requests. One possible solution is the implementation of a chatbot specifically designed to assist in managing and monitoring item stock.

The implementation of a chatbot can be achieved by selecting one of the commonly used algorithms, such as the Pattern Matching algorithm. The Pattern Matching algorithm is employed to search for or match specific patterns within a sequence of characters. This method involves comparing input characters with a predefined pattern to determine if they are present within a given string [6]. In this application, the staff of the Store Department will be the primary users of the chatbot. The chatbot is implemented as a dedicated website for the Store Department staff, enabling users to request the names of items, item codes, quantities, and photos of items that have been input into the cloud database. The website is developed using the Flask framework for the backend, with HTML and CSS used to design the user interface of the chatbot. By leveraging this web-based chatbot,

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 7 No. 1 Januari 2025 Hal. 126-132
http://jurnal.unidha.ac.id/index.php/jteksis
E-ISSN :**2655-8238**
P-ISSN : **2964-2132**

users will be able to manage their time more efficiently and enhance productivity.

Several studies have been conducted using various algorithms applied to chatbots. One such study, titled "Decision Support System for Major Selection in High School Using the Profile Matching Method," aimed to develop a system to assist high school students in choosing their major [7]. Another study, titled "Comparison of the Performance of Random Forest, Xgboost, and Lightbm Algorithms in Classifying Emotions from Reddit Comments," aimed to compare the accuracy of various algorithms, identifying Random Forest as the most accurate. However, utilizing Random Forest requires substantial resources due to its complex construction [8]. A further study, titled "Implementation of String Matching with the Boyer-Moore Algorithm to Determine the Similarity of Thesis Titles (Case Study: XYZ University)" focused on assessing the similarity of thesis titles, though processing larger datasets increases the time required to complete the process, thus necessitating efficient string-matching algorithms for optimal results [9]. Additionally, a study titled "Academic Information Service System Using WhatsApp Chatbot" applied the System Development Life Cycle (SDLC) method for system development [10]. The final related study, titled "Analysis of Chatbot Automation for Administrative Tasks and Management in the Digital Environment Using Python," aimed to create a Python-based chatbot with simplified functionality [11].

This study aims to implement the pattern matching algorithm within a chatbot to support the operations of the Store Department at PT XYZ. The chatbot is designed to perform tasks such as adding data to the database, adjusting warehouse items, displaying item photos, and providing item quantities. The outcomes of this research are expected to establish a basis for selecting the most suitable algorithm for chatbot implementation, thereby optimizing its performance. The development process follows the System Development Life Cycle (SDLC) methodology to ensure that the research is conducted systematically, minimizing errors during the development phase [12]. This study is anticipated to address existing challenges, reduce time-consuming tasks in the Store Department, and enhance the overall efficiency of the work process.

RESEARCH METHODOLOGY

2.1. Research Stages

The research methodology employed in this study follows the System Development Life Cycle (SDLC) as the development approach. The *System Development Life Cycle* (SDLC) is a structured process used in the development of information systems and methodologies for enhancing pre-existing systems [13]. The stages of the research are illustrated in Figure 1.

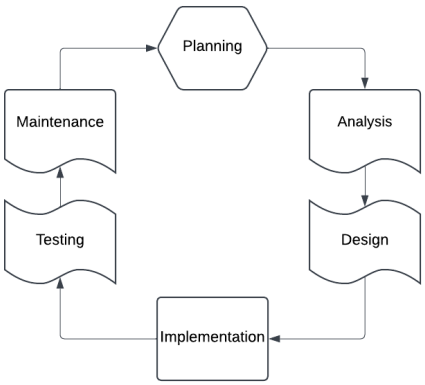

Figure 1 System Development Life Cycle (SDLC)

The first stage of this research is Planning, which aims to establish a research schedule. This stage begins with planning the collection of information deemed useful by the authors for system development, followed by the next stage, which is the Analysis phase [14]. The Analysis phase focuses on identifying the requirements necessary for system development in this research, including the tools and technologies applied to address the identified issues. The next stage is the Design phase, during which the authors design and create a website using Flask as the framework for implementing a Python-based chatbot. For the backend, HTML and CSS are utilized to design the chatbot's user interface. The subsequent stage is Implementation, which involves coding to develop the chatbot. This includes entering data into the chatbot's database and using Python to construct the chatbot with the pattern matching method. The following stage is Testing, a critical step to trial-run the research system and identify any errors or flaws in the chatbot system [15]. Lastly, the Maintenance phase serves as an additional development stage to adapt the system to functional enhancements and meet any new requirements that may arise [16].

2.2 Description of the System

In this study, the system implemented is a Chatbot aimed at generating outputs aligned with user requests. By providing input for a specific item of interest, the chatbot will deliver the corresponding output, including details such as quantity, item code, and an image of the item. If the requested item is not found in the knowledge base or database, the chatbot will notify the user of its unavailability. Furthermore, the chatbot will prompt the user to confirm whether the item should be added to the database. Upon confirmation, the chatbot can

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 7 No. 1 Januari 2025 Hal. 126-132
http://jurnal.unidha.ac.id/index.php/jteksis
E-ISSN :**2655-8238**
P-ISSN : **2964-2132**

automatically incorporate the item into the database. The block diagram design is presented in Figure 2.

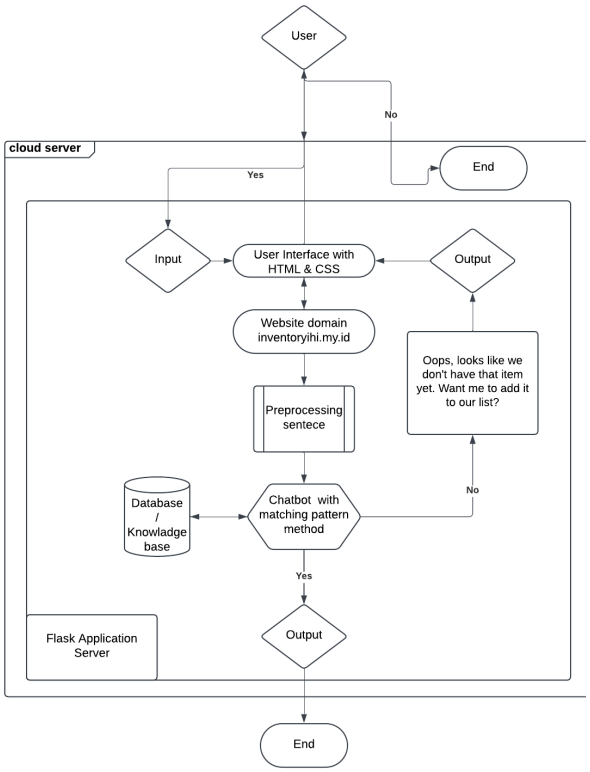

Figure 2 Block Diagram

From Figure 2, it can be observed that the system operates within a cloud server, which hosts a Flask application server as its framework HTML and CSS are utilized as the user interface, serving as a medium for users to provide input and receive output from the chatbot [17][18] User input is first subjected to a preprocessing stage, where the system analyzes the input to identify specific patterns or keywords that may correspond to entries within the knowledge base. This preprocessing is critical for enabling the matching pattern method to effectively search for the requested item in the database [19], If a match is found, the system provides the output, which includes details such as the item's name, code, quantity, or image. Conversely, if no match is identified, the chatbot generates a response informing the user that the requested item is unavailable and offers the option to add the item to the database or terminate the interaction. The Pattern Matching algorithm was selected due to its simplicity, efficiency, and suitability for small to medium-sized datasets, making it ideal for the Store Department's requirements [20].

2.3 System Architecture

The system architecture proposed in this study aims to integrate several key components to support the functionality of a web-based chatbot. The system is initiated by users accessing the service via the internet through the domain inventoryihi.my.id. This domain is hosted on the IDCloudHost platform, which provides the necessary infrastructure to ensure the system's seamless online operation. The hosting platform also serves as the installation environment for the Flask framework, which is employed as the primary tool for website development. Flask facilitates the development process by offering an environment for creating user interfaces utilizing HTML and CSS [21],[22]. Furthermore, Flask serves as the foundation for developing a Python-based chatbot designed to perform pattern matching on data stored in the database. When relevant patterns are identified, the chatbot generates appropriate responses and delivers them to users through the website interface.

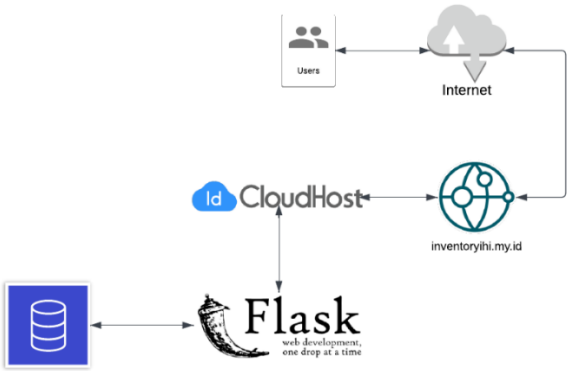

Figure 3 System Architecture

The architectural design proposed in this study is depicted in Figure 3, offering a visual representation of the interconnections among the system's components. The diagram illustrates the integration of users, the domain, hosting services, the Flask framework, and the database, highlighting how these elements collaborate to form a cohesive system. The process begins with user interactions, progresses to the database for processing, and concludes with the delivery of appropriate responses back to the users. This architecture is designed to ensure the system operates with efficiency and responsiveness.

RESULTS AND DISCUSSIONS

3.1 System Implementation

The system development in this study involved the implementation of both system functionality and the design of the user interface. The user interface serves as a platform for users to provide input, which is then processed by the chatbot system using a pattern-matching methodology. The design of the user interface is illustrated in Figure 4.

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 7 No. 1 Januari 2025 Hal. 126-132
http://jurnal.unidha.ac.id/index.php/jteksis

E-ISSN :**2655-8238**
P-ISSN : **2964-2132**

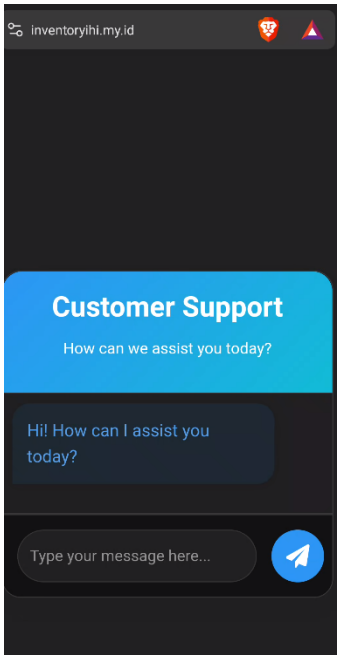

Figure 4 User Inferface

The chatbot processes input by searching for matching strings or patterns within the cloud server database. If the input matches an existing pattern in the database, the chatbot delivers the requested item to the user. The resulting output is displayed in Figure 5.

Figure 5 Output from Chatbot

In cases where no matching string or pattern is found in the knowledge base or database, the chatbot generates a response indicating the unavailability of the requested item. Additionally, the system provides the user with an option to either add a new item or string to the database or decline the addition. This process is depicted in Figure 6.

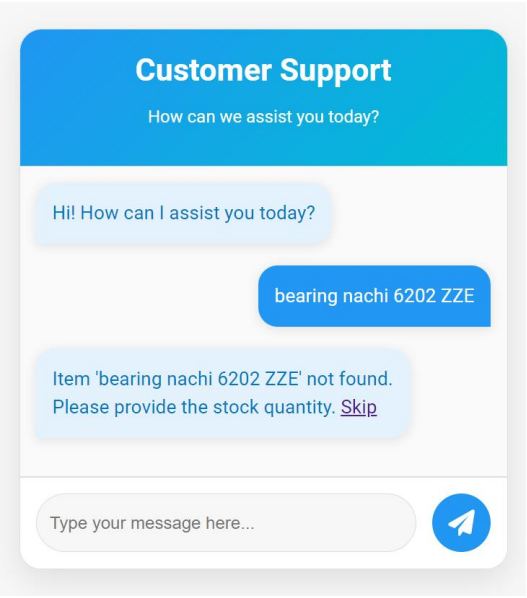

Figure 6 Response for Unavailable Item

If the user chooses to add a new item, they are prompted to input the item's quantity. The new string or item is automatically added to the database, with the exception of the item image, which must be manually uploaded. Once the image is uploaded, the system ensures that the item's name, image, and quantity are displayed correctly during future interactions. The system's responses for this process are presented in Figures 7, 8, and 9.

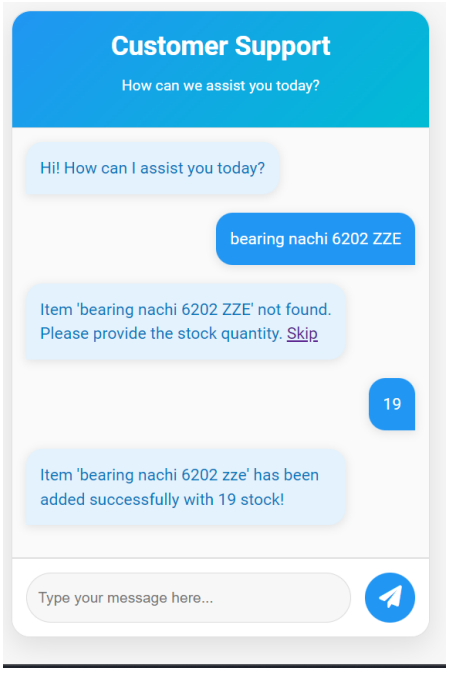

Figure 7 Adding a New Item to the Database

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 7 No. 1 Januari 2025 Hal. 126-132
http://jurnal.unidha.ac.id/index.php/jteksis

E-ISSN :**2655-8238**
P-ISSN : **2964-2132**

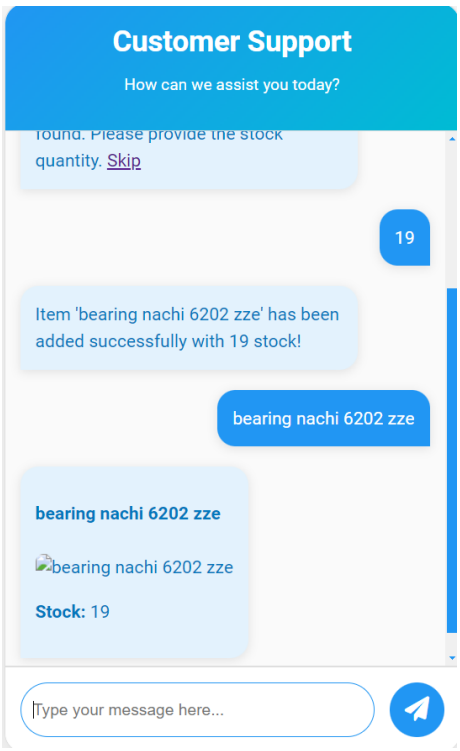

Figure 8 State Before Adding an Image

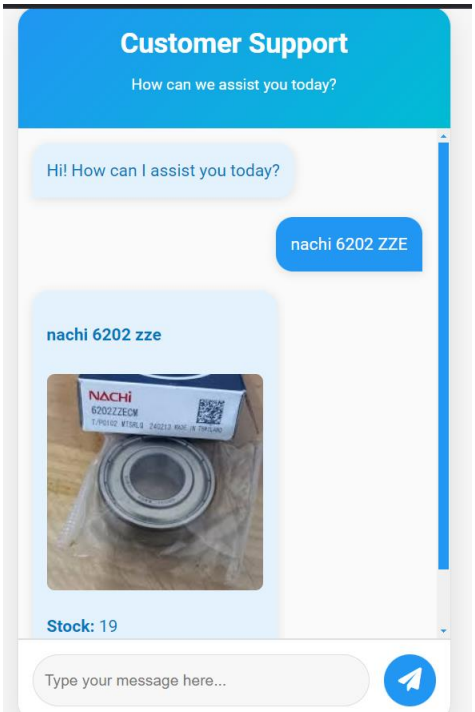

Figure 9 State After Adding an Image

If the user opts not to add a new item or selects the "skip" option, the item addition process is canceled, and the chatbot resumes its operation, allowing the user to search for other items. This scenario is demonstrated in Figure 10.

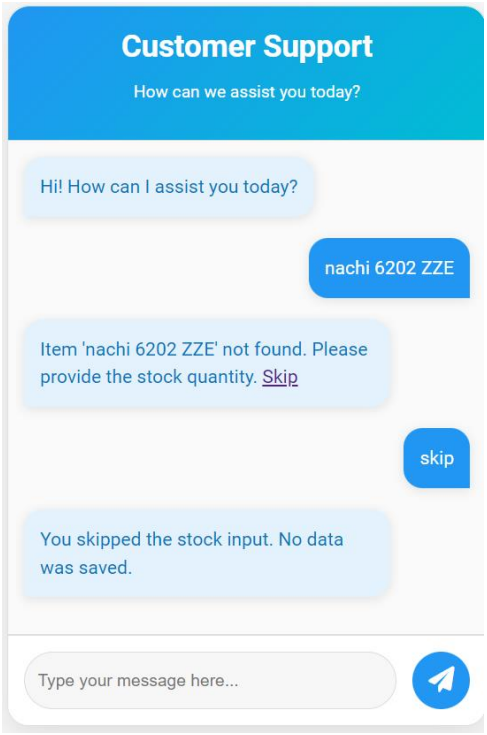

Figure 10 Canceling the Item Addition Process

3.2 Implementation Testing

Black box testing was conducted in this study to determine whether the chatbot can produce the desired outcomes and meet the objectives defined by the authors.

Table 1. Black Box Testing Results

Jurnal Teknologi Dan Sistem Informasi Bisnis
Vol. 7 No. 1 Januari 2025 Hal. 126-132
http://jurnal.unidha.ac.id/index.php/jteksis

E-ISSN : **2655-8238**
P-ISSN : **2964-2132**

Scenario	Result
User Interface	Success
Input Query	Success
Chatbot Output: Input matches pattern in database	Success
Chatbot Output: Input does not match pattern in database	Success
Adding a new string pattern to the knowledge base	Success
Skipping the addition of a new string pattern to the knowledge base	Success

In this black box testing, the evaluation covered various features expected to assist users. These features include the "user interface," "input query," "chatbot output for matching string patterns in the knowledge base," "chatbot output for unmatched string patterns," "adding a new string pattern to the knowledge base," and "skipping the addition of a new string pattern to the knowledge base." By conducting black box testing, the authors were able to assess whether the chatbot system's features functioned as intended to reduce users' workload effectively. The results of the study indicate a "Success" outcome, which satisfied the authors. This demonstrates that the system is ready to be deployed to users, specifically the store department of PT. XYZ.

CONCLUSION

The testing of the web-based chatbot system developed using the pattern matching method demonstrated that the system functions effectively and successfully facilitate users, particularly the store department at PT. XYZ. The system was designed to address user needs, such as adding data to the database, adjusting warehouse items, displaying item images, and providing information about item quantities. By applying the SDLC development methodology, the authors were able to develop the system systematically and in a well-structured manner, resulting in an effective and beneficial chatbot based on the pattern matching method. To evaluate the system's performance, black box testing was employed. The testing results revealed that all features available in the chatbot operated as intended, thereby effectively reducing users' workload, particularly in inventory data management. However, this study still has several limitations that can be addressed in future research. One of the limitations is the use of the pattern matching method, which only supports basic chatbot functionality without advanced AI features. Additionally, the process of inputting images into the database or knowledge base still requires manual intervention by users, which poses a challenge to operational efficiency. As recommendations for future development, the authors suggest integrating machine learning to enhance the chatbot's capabilities, developing features for automating all data input processes, and improving the user interface to make it more intuitive and responsive. With further advancements, the system is expected to provide greater benefits to users and serve as a more sophisticated solution for inventory data management.

REFERENCES

[1] S. Tua and H. Siagian, "Studi komparatif kinerja algoritma pemrosesan bahasa alami dalam sistem Chatbot."

[2] S. Mulyatun, H. Utama, and A. Mustopa, "Pendekatan Natural Language Processing pada aplikasi chatbot sebagai alat bantu costumer service," 2021.

[3] I Gede Suarnata, I Made Sukarsa, and Kadek Suar Wibawa, "Pencocokan menu berbasis keywords pada Chatbot dengan metode Jaccard I Gede Suarnata a1 , I Made Sukarsa a2 , Kadek Suar Wibawa b3."

[4] C. Prianto, R. Andarsyah, and N. H. Harani, "Rancang bangun kamus digital berbasis chatbot menggunakan pendekatan pattern matching," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 6, no. 4, p. 2327, Oct. 2022, doi: 10.30865/mib.v6i4.4910.

[5] K. Aditama, "Pemanfaatan natural language precessing dan pattern matching dalam pembelajaran melalui guru virtual," *ELKOM*, vol. 13, no. 1, pp. 121–133, 2020, doi: 10.51903/elkom.v13i1.187.

[6] R. Ardiansyah, D. Marya, and A. Novianti, "Penggunaan metode string matching pada sistem informasi mahasiswa Polinema dengan chatbot," *JURNAL ELTEK*, vol. 21, no. 1, pp. 28–35, Apr. 2023, doi: 10.33795/eltek.v21i1.381.

[7] Y. Merry Anjani, F. Muttakin, and I. Permana, "Sistem Pendukung Keputusan Pemilihan Jurusan pada SMA menggunakan Metode Profile Matching," 2024, doi: 10.47065/josyc.v5i3.5166.

[8] F. D. U. Arif, "Perbandingan kinerja algoritma Random forest, Xgboost dan Lightgbm dalam klasifikasi emosi komentar Reddit," Jakarta, Sep. 2023. Accessed: Aug. 12, 2024. [Online]. Available: https://repository.uinjkt.ac.id/dspace/handle/123456789/76259

[9] I. Ahmad, R. Indra Borman, G. G. Caksana, and J. Fakhrurozi, "SINTECH Journal | 53 IMPLEMENTASI STRING MATCHING DENGAN ALGORITMA BOYER-MOORE UNTUK MENENTUKAN TINGKAT KEMIRIPAN PADA PENGAJUAN JUDUL SKRIPSI/TA MAHASISWA (STUDI KASUS: UNIVERSITAS XYZ)", [Online]. Available: https://doi.org/10.31598

[10] A. Rizaldy, A. Muhammad Nur Hidayat, I. Permata Dwitami, and K. Kunci, "SYNCTECH VOL. 1 NO. 1 (2024) XXXX-XXXX (EISSN) Sistem Layanan

E-ISSN :**2655-8238**
P-ISSN : **2964-2132**

Informasi Akademik Menggunakan Chatbot Whatsapp."

[11] Christian Iwan, Christvaldo Kurnia Putra, Dianeluoxy Zabdi, Elson Ivan Boy, Monica Agustina Chandra, and Lifia Yola Febrianti, "Analisis pemanfaatan artificial intelligence dalam membantu proses perekrutan karyawan perusahaan," *Jurnal Sains dan Teknologi*, vol. 2, no. 2, pp. 161–168, Dec. 2023, doi: 10.58169/saintek.v2i2.248.

[12] K. Nurul Musthofa *et al.*, "Perancangan sistem informasi absensi dan permohonan cuti karyawan berbasis web menggunakan metode System Development Life Cycle (SDLC) pada SD Budi Mulia Dua Bintaro," *JORAPI : Journal of Research and Publication Innovation*, vol. 1, no. 3, 2023, [Online]. Available: https://jurnal.portalpublikasi.id/index.php/JORAPI/index

[13] W. Erawati, S. Heristian, R. A. Purnama, and C. Author, "Rancang bangun sistem informasi akademik berbasis website dengan metode SDLC," Jakarta, Jul. 2023. doi: 10.31294/coscience.v3i2.1918.

[14] I. Ruslianto and Y. Erniajan, "Penerapan Model Waterfall dalam Pengembangan Perangkat Lunak Pemantauan Tanaman Anggur Berbasis Mobile Menggunakan IoT," 2024, doi: 10.47065/josyc.v5i3.5099.

[15] R. Ranti Rosalina, A. Fasha Madhani, and R. Suwartika Kusumadiarti, "Perancangan Sistem Informasi Akuntansi Pembelian Bahan Baku Kredit Berbasis Web Pada PT XYZ," *Jurnal E-Bis*, vol. 8, no. 2, pp. 882–902, Oct. 2024, doi: 10.37339/e-bis.v8i2.1985.

[16] D. Mallisza, H. S. Hadi, and A. T. Aulia, "Implementasi Model Waterfall Dalam Perancangan Sistem Surat Perintah Perjalanan Dinas Berbasis Website Dengan Metode SDLC," *Jurnal Teknik, Komputer, Agroteknologi Dan Sains*, vol. 1, no. 1, pp. 24–35, Jun. 2022, doi: 10.56248/marostek.v1i1.9.

[17] D. Kusnadi and E. R. Yulia, "Sistem Informasi Program Stock Opname Berbasis Website," 2023. [Online]. Available: http://jurnal.bsi.ac.id/index.php/imtechno

[18] A. Fisryansah Ahlief Putra, S. Maarif Aceh, A. Nasution, and M. Syahuda Hasibuan, "Pengembangan Chatbot Sederhana Untuk Informasi Akademi Menggunakan Python Dan Flash Untuk Masyarakat."

[19] M. Zafar, "Developing Smart Conversation Agent ECOM-BOT for Ecommerce Applications using Deep Learning and Pattern Matching," *International Journal of Information Engineering and Electronic Business*, vol. 15, no. 2, pp. 1–10, Apr. 2023, doi: 10.5815/ijieeb.2023.02.01.

[20] G. Nivedhitha, E. Punarselvam, K. R. Aaghash, M. Elayabarathi, and R. S. K. Rahul, "Ai Consulting Healthcare Chatbot System Using Pattern Matching." doi: 10.32628/IJSRST2182112.

[21] A. Yulianto and E. Lau, "Development of an Integrated Chatbot on the Website Using IBM Watson Assistant."

[22] A. Yulianto, "Prosiding National Conference for Community Service Project (NaCosPro)", doi: 10.37253/nacospro.v5i1.8065.