

DEVELOPMENT OF CHATBOT SERVICES FOR ORDERING MEDIA SUPPORT USING FUZZY LOGIC ALGORITHM: CASE STUDY OF PT. PEMUDA CARI CUAN (MANGKOKKU)

^{1*}ARIS AFRIYANTO, ²YOHANES EKA WIBAWA

Department of Informatics Engineering, Tanri Abeng University

Jl. Swadarma Raya No.58, Ulujami, Kec. Pesanggrahan, Kota Jakarta Selatan

e-mail: ¹aris@student.tau.ac.id, ²yohanes.eka@tau.ac.id

*Corresponding author

ABSTRACT

The importance of customer service in the modern business world cannot be ignored. Providing customers with good service can increase their satisfaction and strengthen the company's and customers' relationship. In the digital era, customer service increasingly shifts to online platforms, such as social media and instant messaging applications. However, providing efficient and responsive services on these platforms can take time and effort. At Mangkokku Restaurant, using social media platforms for promotions, discounts, and other information has become part of the business strategy. However, many customers need help understanding the context of the information in Mangkokku's social media posts. In addition, customers at dine-in outlets often repeat questions about promotions and discounts to the staff at the outlet, especially when the outlet is busy, and the staff needs help providing excellent and fast service. Therefore, research on chatbot services is considered a solution to overcome limitations in communication between staff and customers. With technological advances, chatbots are expected to provide positive benefits. This research uses a fuzzy logic algorithm to help chatbots find the expected response from customers. Apart from that, interviews with the marketing division, customer service, and outlet staff were conducted to collect data on questions and information that needed to be conveyed to customers. In this way, customers at Mangkokku Restaurant are expected to be able to quickly and efficiently ask about menus, promos, discounts, or other information via this chatbot service.

Keywords: *Customer service, Chatbot, Fuzzy Logic, Mangkokku Indonesia*

1. INTRODUCTION

Mangkokku is a company engaged in the field of Food & Beverage (Restaurant) made by Randy Julius Kartadinata, Chef Arnold Poernomo, Gibran Rakabumingraka, Kaesang Pangarep with the tagline No.1 Chef Quality Comfort Food which serves a variety of delicious and comfortable food combined with a variety of typical Indonesian chili sauce. Mangkokku currently serves dine-in, takeaway, and online. Online services are available at several Gofood, Grabfood, Shopeefood, and WhatsApp merchants. However, service to customers regarding questions about menus, promos, discounts, and other attractive offers is currently only through the working staff's social media. If the outlet is busy and many orders, customers who contact Mangkokku via social media or through staff at the outlet experience slow response and are not well served.

Previous research becomes the foundation for further research. Its usefulness is to explain the results of previous researchers. The following previous research is a reference for this research, including the design of customer service chatbot features using Dialogflow [1]. This system can replace the role of humans in customer service tasks. One of them is customer service, answering customer questions. Further research based on the results of problem formulation to the implementation stage can be concluded that the use of chatbots is a solution to the problems faced in the existing system. There is a Text Preprocessing process to reduce errors in the input identification process with fuzzy logic algorithms.

Several studies related to restaurant services have been carried out previously, including a hotel reservation system via Facebook Messenger with satisfactory results [2], the use of Open-AI in Indian food ordering to provide service with chatbots [3], online food ordering application at food stalls using buyer location information [4], Grill and Suki food ordering system using Scrum [5], restaurant information system to facilitate food and room reservations [6], restaurant reservations using mobile applications [7]. Chatbot research also continues to be developed, such as for elementary school student learning [8], for academic information system services on campus [8], for improving student services [9], for campus customer service [10], and as a server in new student admissions information systems [11]. Previous research focused on using chatbots as part of information systems in general, but there has yet to be any development with a combination of fuzzy logic. This combination will become a more humanistic system because it absorbs human fuzzy reasoning.

The existence of this research is expected to overcome the current problems, namely serving customers by providing menu product information, discount promos, and other exciting things by developing Telegram social media-based chatbot services using the Python programming language with fuzzy logic algorithms as an approach used in chatbot service programming.

2. RESEARCH METHODOLOGY

2.1 Flowchart Research

The methods used to collect data directly include. The author uses the bibliographic method to search for information on the internet, and the research and findings are very relevant to the problem being discussed. This approach is used as a reference in the writing process. Field methods involved direct observation of branch staff interactions with customers and introductions at the Mangkokku representative office. This field study includes data collection methods such as. Interview: The author conducted a question and answer session with outlet staff and Mangkokku divisions related to customer service. In this study, researchers asked various questions about the customer service system at the Mangkokku outlet. This method is used to collect diverse information. Observation: This method allows the author to directly witness the customer service process at the Mangkokku outlet. The results generate data and insights into the customer service system in operation.

2.2 System Development Methods

This research adopts the SDLC (Software Development Life Cycle) model for system development, using the Waterfall model. Waterfall Model or Classic Life Cycle is a commonly used model in software engineering. This model is called "waterfall" because each step must wait for the completion of the previous step and is executed sequentially. The following are the steps in developing chatbot services using the Waterfall system development method

1. Requirement Analysis

In this phase, system developers need to communicate with the aim of understanding the user's expectations of the software and the limitations that may exist. This information is generally collected through interviews, discussions, or direct surveys. The data is then analyzed to provide relevant information to the user.

2. System Design Specification

In this stage, the requirements from the previous stage are described in detail and the system design is prepared. The system design helps in specifying hardware and software requirements, and contributes to defining the overall system architecture.

3. Implementation

In this step, the system is first built through small programs known as integrated units, and the process of developing and testing the functionality of each unit is referred to as unit testing.

4. Integration & Testing

After each unit test, all components developed in the implementation stage are integrated into the system. Once the integration process is complete, the entire system is tested to detect any potential failures or errors.

5. Operation & Maintenance

The final step of the waterfall approach and maintenance of the completed software includes bug fixes that were not detected in the previous stages. New requirements involve improving the implementation of system units and improving system services.

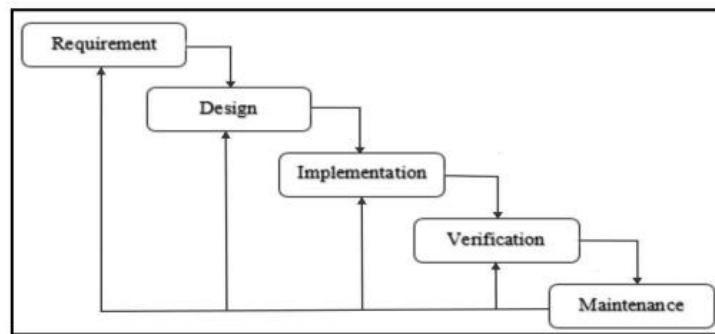


Figure 1. Waterfall Method

2.3 Chatbot

Chatbot consists of two words, “chat” and “bot”. This technology is a form of artificial intelligence created to communicate automatically between humans and machines without human intervention. A chatbot plays the role of a bot in serving conversations. It is important to note that despite being called “bots”, these chatbots are not actual physical robots, but rather specially designed applications [12].

The development of a chatbot can be customized to a specific function or need, and each type of chatbot developed has specific characteristics to handle questions from users [13]. The types of chatbots are as follows:

1. Button-Based Chatbot

Button-Based Chatbot is a type of chatbot whose user input is based on the command buttons provided by the chatbot. In this button-based chatbot, the knowledge base created by the developer is not too large, because user input is only based on the buttons provided by the chatbot system. The downside of this type of chatbot is that each button provides a query for input, so users cannot freely provide input or questions.

2. Keyword Recognition-Based Chatbot

Keyword Recognition-Based Chatbot is a type of chatbot that responds to user input by recognizing some keywords in user input. The input received by this type of chatbot is in the form of a sentence or series of words. This type of chatbot determines the appropriate response or responses to the input given by examining each word until it finds the keywords that match the knowledge base in the system.

3. Contextual Chatbot

Contextual Chatbot is a type of chatbot that uses machine learning and artificial intelligence to handle the system's conversation with the user. A contextual chatbot when receiving input is the same as a keyword recognition-based chatbot. That is, it accepts input consisting of words or sentences. The difference between contextual chatbots and other types of chatbots is that all user input is stored and double-checked, so the system can respond appropriately to input that is worded differently but has the same meaning. Contextual chatbots require the development of a knowledge base that must be developed regularly in order to learn from each user input.

2.4 Customer Service

Customer service, or CS for short, is a type of service provided by companies to customers before or after the purchase of a product or service. Therefore, Customer Service is a division that plays an important role in company protection and has direct interaction with customers. Although sometimes considered trivial, work in customer service involves many tasks and roles within the company.

The two main roles of customer service are being a chief PR officer and creating a positive company culture. Customer service can also be defined as direct communication between consumers who make purchases and personnel representing the brand or company. The focus of customer service is to provide convenience to improve the transaction experience, with the aim of meeting customer expectations. Good customer service quality is one of the key factors in the progress of a company. Over time, forms of customer service have become more flexible. If previously customer service was done over the phone, now many websites provide customer service via chat. Not only that, customer service can also be done via email, SMS, and even through social media.

2.5 Python and Telegram Bot

Python is a programming language created by Guido van Rossum, a native of Amsterdam, Netherlands. This programming language is famous for its ease of learning and understanding by various groups. Python's data structures

have a high level of complexity and use a simple yet effective approach in the development of object-oriented programming [4].

Telegram provides bots, which are specialized Telegram accounts designed to respond automatically to specific text commands by providing pre-formatted responses. Interaction with bots can be done by users through sending command messages in both private and group messages [14]. The process of creating and setting up bots can be done using Botfather, which is a Telegram bot agent that sets the rules for each bot. Botfather assists users in creating new bots and configuring the settings of existing bots. Communication between the bot and the developer's server can be done through an API provided by Telegram, the Bot API. In addition to giving developers the freedom to customize bot responses, Telegram also provides bot libraries in various programming languages, including Python using the pyTelegramBotAPI library. Bots can perform various functions, such as sending customized notifications, creating specialized tools (such as weather forecasts, translators, etc.), creating games, and various other tasks.

2.6 Telegram Bot Api

Bot API which is an HTTP-based interface specifically designed for developers interested in creating bots on the Telegram platform. With the Bot API, developers can establish communication between bots on Telegram and developer servers. Each bot has its own unique token, which serves as the key to accessing the bot and controlling the developer server. All interactions with the Bot API must be done via the HTTPS protocol at the address https://api.telegram.org/bot<token>/METHOD_NAME. In addition to using Telegram servers, the server communication option also involves building a local server. When a bot request is sent to a local server, no HTTPS format check is required as it happens on Telegram servers. The use of local servers provides several advantages, including unlimited file upload capabilities and support for files up to 2000 MB in size.

2.7 Fuzzy Logic

In a linguistic context, the term “fuzzy” is defined as something vague or imprecise, indicating that a value can be considered both true and false simultaneously. Fuzzies have membership degrees that range from 0 (zero) to 1 (one). Fuzzy logic is a form of logic that involves vagueness or ambiguity between truth and error. In fuzzy logic theory, a value can have both truth and error at once, and how much it is true or false depends on the membership weight. Membership degrees in fuzzy logic are measured from 0 to 1, and fuzzy logic describes the extent to which a value is considered true and the extent to which it is considered false. Fuzzy logic presents an easy method to map inputs to outputs with continuous values. Fuzziness is expressed through the concept of membership and degrees of truth. Therefore, it can be concluded that something at the same time has a part that is considered true and another part that is considered false [15].

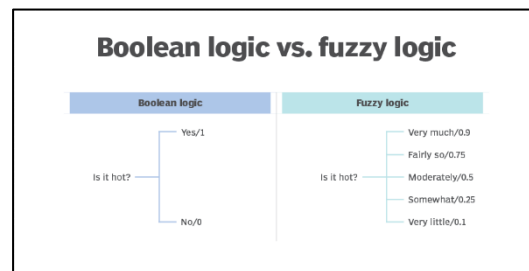


Figure 2. Comparison between boolean logic and fuzzy logic

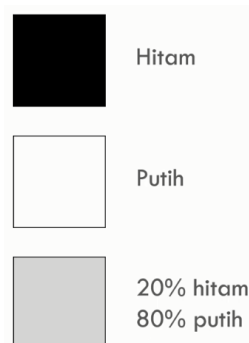


Figure 3. Example of fuzzy logic algorithm in color comparison

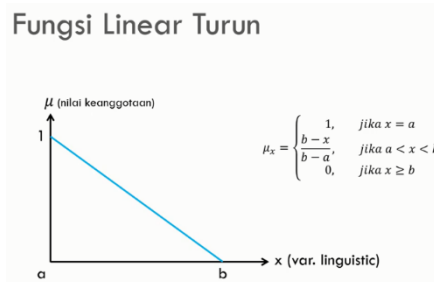


Figure 4. Fuzzy Logic Formula Linear Descending Function

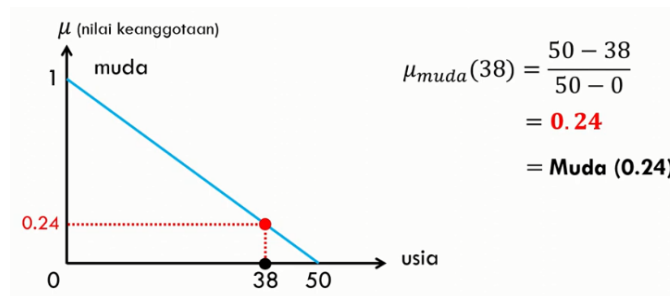


Figure 5. Example of Fuzzy Logic Calculation of Linear Descent Function in Age Comparison

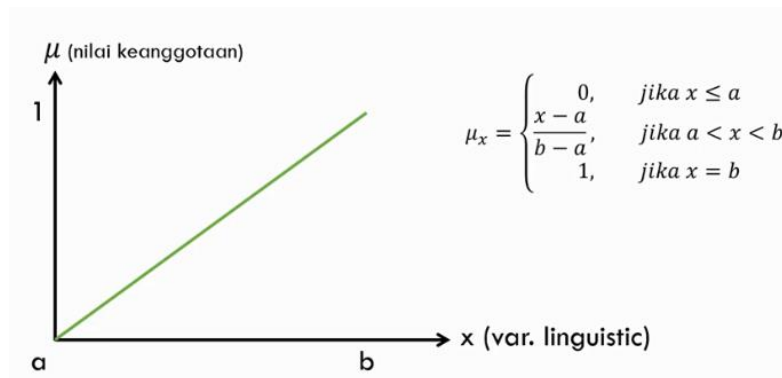


Figure 6, Fuzzy Logic Formula Linear Ascending Function

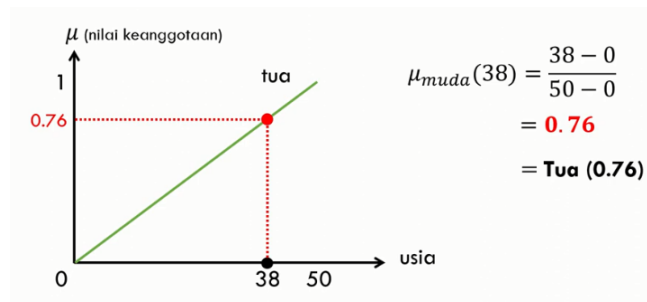


Figure 7. Example of Upward Linear Fuzzy Logic Calculation in Age Comparison

2.8 FuzzyWuzzy

FuzzyWuzzy is a Python library used to perform string matching using a fuzzy approach. Basically, the concept of fuzzy string matching aims to find strings that are similar to a certain pattern, even if they are not exactly the same. This library utilizes Levenshtein distance to measure the difference between strings in a character set. FuzzyWuzzy provides several methods to compare strings, including FuzzyWuzzy Ratio, FuzzyWuzzy PartialRatio, FuzzyWuzzy

TokenSortRatio, FuzzyWuzzy TokenSetRatio, and FuzzyWuzzy WRatio. Each method has advantages and disadvantages depending on user requirements. When used, FuzzyWuzzy generates a score between 0 and 100, which reflects the degree of similarity between the two strings being compared; the higher the score, the more similar the two strings are.

3. RESULT AND DISCUSSIONS

3.1 User Process Overview

Users must have and login to Telegram social media account to be able to use the chatbot system.

3.2 System Development and Running Process

In the process of running the system, all input from customers is processed with a fuzzy logic algorithm to determine the appropriate response. Chatbot Service with Waterfall System Development Method as follows:

- Formulating the design method: In this step, the author applies the waterfall method for sequential and effective system development. It involves planning the materials required by the system and collecting data.
- Making UML Design: At this stage, the author makes UML (Unified Modeling Language) designs including use case diagram and activity diagram.
- Creating an interface design: At this stage, the author designs the user interface for the chatbot service system.
- Implementation of the design and coding process: At this stage the author begins coding the chatbot service system that is built. In addition, there is also an architectural design, the system architecture design aims to describe the overall process of how the chatbot system can run. For example, requesting a Telegram bot API token so that the chatbot can run on Telegram social media. Customers will send text writing requests or questions to the chatbot in the Telegram social media application. Then the chatbot will reply according to what the customer asked. The following flow of how to deploy a chatbot service system using the telegram bot API using Python anywhere as a chatbot server can be seen in Figure 11.
- Testing the chatbot system: At this stage the author tests the chatbot service system, whether the system is running well and ensures that the system can be used by customers. As an illustration of the process of running the chatbot service system can be seen in Figure 12, 13, 14, 15.

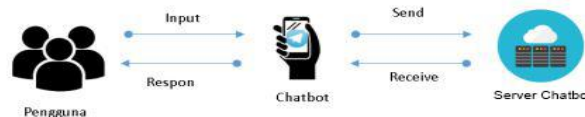


Figure 8. General Description of Telegram Chatbot System

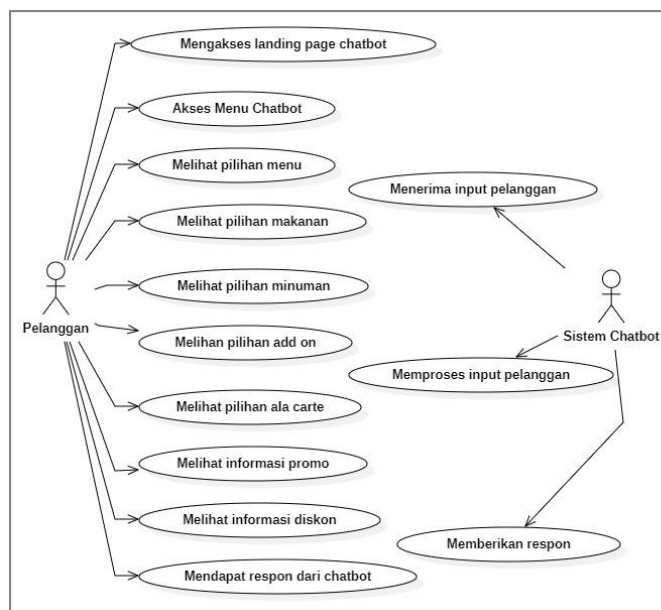


Figure 9. Use case diagram of chatbot service system

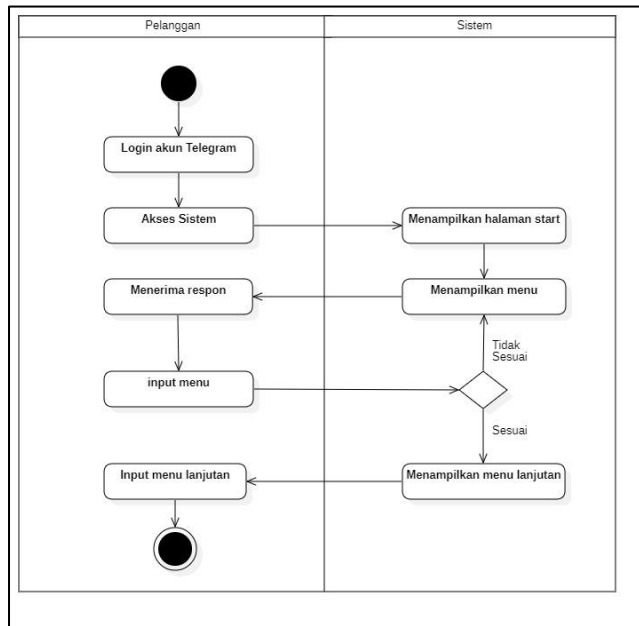


Figure 10. Activity diagram of the system

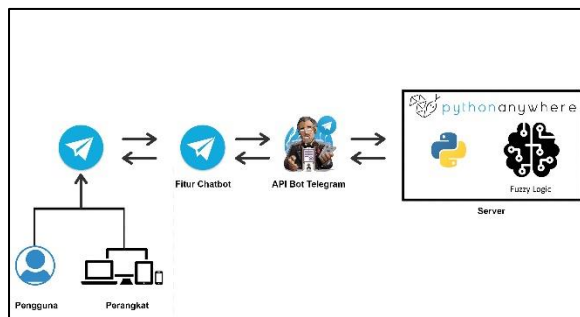


Figure 11. Chatbot system architecture design

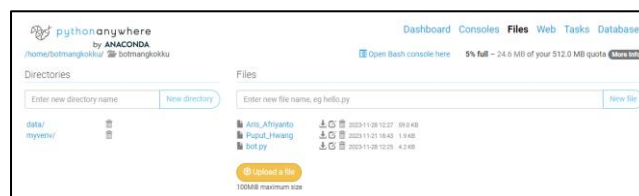


Figure 12. Cloud Server Pythonanywhere



```
1 import telebot
2 import random
3 import requests
4 import datetime
5 import fuzzywuzzy
6 import pytz
7 import time
8 import json
9 import csv
10
11 from fuzzywuzzy import process
12
13 token = '6176037546:AAHckZneFwB5Eb460jehDRuP0FsLxPKFw0o' # Token API Bot Telegram
14 local_timezone = pytz.timezone("Asia/Jakarta") # Update Timezone WIB
15
16 class BotHandler:
17     def __init__(self, token):
18         self.token = token
19         self.api_url = "https://api.telegram.org/bot{}/".format(token)
20
21     def get_updates(self, offset=None, timeout=30):
22
23 start 5 <class 'str'>
24 ('start', 100)
25 LAST UPDATE: <class 'dict'>
26 {'update_id': 384565251, 'message': {'message_id': 640, 'from': {'id': 5667502343, 'is_bot': False, 'first_name': 'Aris', 'last_name': 'Afriyanto', 'username': 'ariezafrian', 'language_code': 'en'}, 'chat': {'id': 5667502343, 'first_name': 'Aris', 'last_name': 'Afriyanto', 'username': 'ariezafrian', 'type': 'private'}, 'date': 1701364942, 'text': 'mau tanya menu'}}
27 mau tanya menu 14 <class 'str'>
28 ('Menu', 90)
29 LAST UPDATE: <class 'dict'>
30 {'update_id': 384565252, 'message': {'message_id': 642, 'from': {'id': 5667502343, 'is_bot': False, 'first_name': 'Aris', 'last_name': 'Afriyanto', 'username': 'ariezafrian', 'language_code': 'en'}, 'chat': {'id': 5667502343, 'first_name': 'Aris', 'last_name': 'Afriyanto', 'username': 'ariezafrian', 'type': 'private'}, 'date': 1701364951, 'text': 'boleh lihat makanan'}}
31 boleh lihat makanan 19 <class 'str'>
32 ('Food, Makanan', 70)
```

Figure 13. Testing the chatbot system on Pythonanywhere Cloud Server

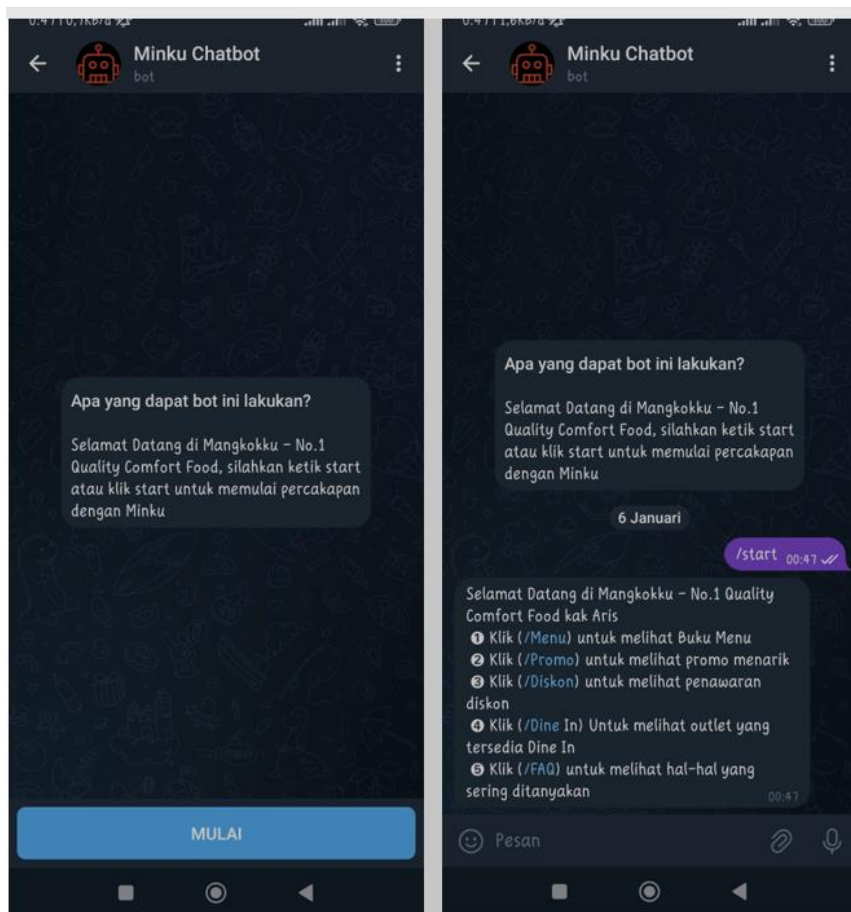


Figure 14. Interface Implementation

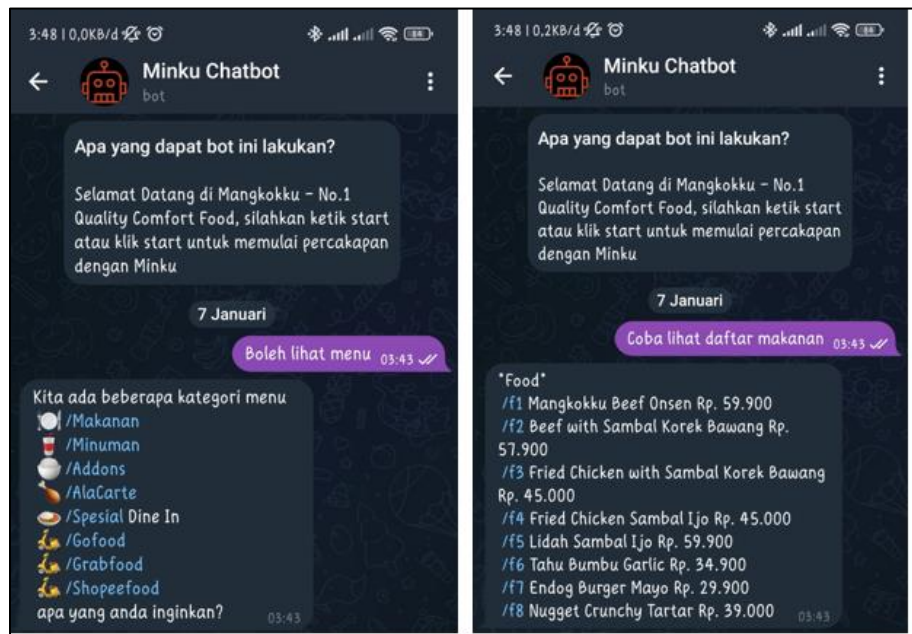


Figure 15. Interface Implementation

The test results, presented in Figure 14-16, show that the system can provide answers according to the buyer's order context. This way, the system has worked well and can be implemented in the real world.

4. CONCLUSION

The conclusions obtained based on the discussion show that development of a chatbot service system using the bot API from Telegram social media, using the python programming language combined with a fuzzy logic algorithm approach. This chatbot service system was created using the SDLC Waterfall methodology running on the Pythonanywhere python cloud server. With the chatbot service system as a support for ordering media at Mangkokku Indonesia, it can improve service to customers quickly, efficiently and also make it easier for customers to find menu information, promos, discounts and other attractive offers offered at Mangkokku Indonesia.

REFERENCES

- [1] J. Wiratama, S. A. Sanjaya, and V. I. Sugara, (2022), "Rancang Bangun Fitur Chatbot Customer Service Menggunakan Dialogflow," *Komputasi: Jurnal Ilmiah Ilmu Komputer dan Matematika*, vol. 19, no. 1, pp. 25–37, doi:10.33751/KOMPUTASI.V19I1.4474.
- [2] M. S. Haq, B. Wasposito, and N. Nuryasin, (2021), "Chatbot: Reservasi Restoran Online pada Facebook Messenger," *Applied Information System and Management (AISM)*, vol. 4, no. 1, pp. 23–30, doi:10.15408/aism.v4i1.19765.
- [3] N. P. Florensia *et al.*, (2024), "Implementasi Open AI Pada Website Restoran Makanan India," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 3, pp. 3766–3772, doi:10.36040/JATI.V8I3.9770.
- [4] A. fatima Zahra and Iwan Setiawan Wibisono, (2023), "Aplikasi Pemesanan Makanan Online Berbasis Chatbot Di Warung Makan Barokah Kab. Semarang," *Jurnal Mahasiswa Teknik Informatika*, vol. 2, no. 1, pp. 46–55, doi:10.35473/jamastika.v2i1.1870.
- [5] A. P. Nurda and Petrus Sianggian Purba, (2022), "Perancangan Sistem Informasi Reservasi & Makanan Pada Restoran Moji Grill & Suki Berbasis Web Menggunakan Metode Pengembangan Scrum," *Jurnal Publikasi Teknik Informatika*, vol. 1, no. 3, pp. 42–61, doi:10.55606/juhti.v1i3.607.
- [6] A. F. Bahri, (2022), "Sistem Informasi Manajemen Reservasi Restoran Dan Penyewaan Ruangan Berbasis Mobile (Studi Kasus: Begadang Resto)," *Jurnal Teknologi dan Sistem Informasi*, vol. 3, no. 4, pp. 28–33.
- [7] Y. D. Sirait, (2021), "Aplikasi Reservasi Menu Restoran Berbasis Web dan Mobile Android di Cowek Ireng," *Jurnal Sistem Informasi Dan Teknologi (SINTEK)*, vol. 1, no. 2, pp. 1–6.
- [8] R. Parina, A. Wijaya, and Y. Apridiansyah, (2022), "Aplikasi Chatbot Sebagai Media Pembelajaran Interaktif SD N 17 Kota Bengkulu Berbasis Android," *Jurnal Media Infotama*, vol. 18, no. 1, p. 121.

- [9] E. Sabna, (2022), “Aplikasi Chatbot Sebagai Customer Support Untuk Meningkatkan Pelayanan Terhadap Calon Mahasiswa,” *Jurnal Ilmu Komputer*, vol. 11, no. 1, pp. 21–24, doi:10.33060/jik/2022/vol11.iss1.249.
- [10] H. Herman, A. Yulianto, and S. Lim, (2023), “Analisa Aplikasi Chatbot Untuk Pertanyaan Umum Terkait Universitas Di Universitas Xyz,” *Jurnal Pendidikan Teknologi Informasi (JUKANTI)*, vol. 6, no. 2, pp. 415–437, doi:10.37792/jukanti.v6i2.1069.
- [11] G. Guntoro, Loneli Costaner, and L. Lisnawita, (2020), “Aplikasi Chatbot untuk Layanan Informasi dan Akademik Kampus Berbasis Artificial Intelligence Markup Language (AIML),” *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, vol. 11, no. 2, pp. 291–300, doi:10.31849/digitalzone.v11i2.5049.
- [12] J. Enterprise, *Teknik Memasang Chatbot di Toko Online*. Jakarta: Elex Media Komputindo, 2018.
- [13] I. I. Ramli, “Rancang Bangun Aplikasi Chatbot Penyedia Informasi Bengkel Menggunakan Algoritma Nazief & Adriani (Studi Kasus: Bengkel SUM Hexaputra).” Universitas Multimedia Nusantara, Tangerang, 2018.
- [14] R. Nufusula and A. Susanto, (2018), “Rancang Bangun Chat Bot Pada Server Palsa Menggunakan Telegram Bot API,” *JOINS (Journal of Information System)*, vol. 3, no. 1, pp. 80–88, doi:10.33633/JOINS.V3I1.1884.
- [15] I. Raga Djara, T. Widiastuti, and D. M. Sihotang, (2019), “Penerapan Logika Fuzzy Menggunakan Metode Mamdani Dalam Optimasi Permintaan Obat,” *Jurnal Komputer dan Informatika*, vol. 7, no. 2, pp. 157–161, doi:10.35508/jicon.v7i2.1645.