DESIGN OF CLOUD-BASED CHATBOT APPLICATION AT PT. TRAVELOKA SINGAPORE USING THE AGILE METHOD

Yuda Adi Pratama[1], Titin Kristiana[*)]

Information System
Nusa Mandiri University
11211134@nusamandiri.ac.id, titin.tka@nusamandiri.ac.id
yudaa0110@gmail.com, titin.tka@nusamandiri.ac.id
(*) Corresponding Author

Abstract— The role of customer service will be good if it can serve all obstacles or difficulties users face directly and in real time. However, there are times when the number of customer service is not proportional to the number of users who make complaints to customer service, and often, users ask questions generally available on Frequent Ask Questions (FAQ), so there are limitations and are fewer responsive in serving user complaints. By developing a cloud computing-based Chatbot application, it is hoped that it will make it easier for customer service to handle recurring questions and increase response time to users in real-time. The development of this chatbot application uses the agile method with the scrum framework. Where in the development process carried out is divided into several phases called sprints. The development of this application was carried out in 3 sprints from the time the project was announced to completion.

Keywords: Chatbot, Cloud Computing, Agile.

Abstrak— Peran customer service akan baik jika dapat melayani segala kendala atau kesulitan yang dihadapi pengguna secara langsung dan real-time. Namun ada kalanya jumlah customer service tidak sebanding dengan jumlah user yang melakukan komplain ke customer service dan seringnya user mengajukan pertanyaan yang umumnya tersedia pada Frequent Ask Questions (FAQ) sehingga terjadi keterbatasan dan kurang tanggap dalam melayani keluhan pengguna. Dengan mengembangkan aplikasi Chatbot berbasis cloud computing, diharapkan akan memudahkan customer service dalam menangani pertanyaan berulang dan meningkatkan waktu respon kepada pengguna secara real-time. Pengembangan aplikasi chatbot ini menggunakan metode agile dengan framework scrum. Dimana dalam proses pengembangan yang dilakukan dibagi menjadi beberapa fase yang disebut dengan sprint. Pengembangan aplikasi ini dilakukan dalam 3 sprint sejak proyek diumumkan hingga selesai.

Kata Kunci: Chatbot, Cloud Computing, Agile.

INTRODUCTION

Traveloka Singapore is a leading travel and lifestyle platform that offers various services to help users plan their trips and book flights, hotels, activities, and other travel-related services. With the increasing number of users and features on the Traveloka Singapore application, providing quality customer service has become crucial to enhancing user experience and satisfaction.

Effective customer service plays a significant role in building solid customer relationships and increasing loyalty toward Traveloka Singapore. Users expect fast and appropriate solutions to their queries, complaints, or problems. However, sometimes, the number of customer service personnel may not be sufficient to cater to the growing user base, resulting in delayed responses and unsatisfied customers (Çallı and Çallı 2022; Taylor et al. 2020). In addition, users may also ask repetitive questions already available in the Frequently Asked Questions (FAQ) section, leading to inefficiency in resolving customer queries.

Building a cloud computing-based chatbot application aims to make it easier for customer services to handle recurring user questions and complaints in real-time (Li and Zhang 2023). One of the objectives is to facilitate users submitting complaints to customer services by providing a chatbot that can respond quickly and efficiently to their concerns. Additionally, the chatbot will be able to handle repetitive questions and complaints, improving response time to users. As a result, users will not have to wait for long periods before their issues are resolved. Finally, the chatbot will help to reduce the operational burden of customer services. This is because the chatbot can handle many tasks typically performed by human customer service representatives, such as responding to frequently asked questions and resolving common issues.

Studies by (Ananda Dwi et al. 2018) and (Nugroho, Adi, and Gumelar 2020) have highlighted the need for innovative solutions to address these challenges. One such solution is integrating chatbot technology, powered by cloud computing, to facilitate customer service operations. Chatbots are

PILAR Nusa Mandiri: Journal of Computing and Information System
Vol. 19, No. 1 March 2023 | DOI: 10.33480/pilar.v1i1.3055

intelligent conversational agents that can automate customer service processes by answering frequently asked questions, providing real-time assistance, and even resolving simple issues without human intervention (Enterprise 2018).

The previous research from S. Sudaryono, N. Lestari, and K. Gunawan (Sudaryono, Lestari, and Gunawan 2020) shows that building a Virtual Assistant significantly impacts human work and provides quick and easy access to information. The Agile concept with the Scrum method used in the development of Virtual Assistants makes it fast and efficient, which enables the application to provide excellent service to all its users. Meanwhile (Chandra and Kosdiana 2019) indicates that building the Chatbot Line application can improve the efficiency and speed of responding to user communications and providing information.

Based on the journal of (Ananda Dwi et al. 2018) concludes that the chatbot application built can help SME Minsu's customer service by answering customer questions and placing orders for products. The chatbot can serve the role of human customer service efficiently and effectively. Another journal from (Astuti and Fatchan 2019) shows that building a Chatbot application called RiChat simplifies customer service operations in answering user questions, making it more efficient, effective, and faster since the chatbot can work 24 hours a day.

Although no GAP analysis was conducted in this study, the author attempts to explore the topic by conducting a comprehensive literature review and in-depth analysis of the problem to be solved. The author found several studies related to the same topic in the literature review but with different approaches and methodologies. Therefore, the author strives to adopt the appropriate approach and methodology to address the problem.

Using cloud computing technology, Traveloka Singapore can optimize its customer service operations by deploying a chatbot application that can address frequently asked questions and resolve simple issues in real time. This approach can reduce the workload of customer service personnel and enable them to focus on complex queries and complaints that require human intervention (Larasati Amalia and Wahyu Wibowo 2019). The chatbot application can also provide 24/7 customer support, ensuring users receive prompt assistance whenever required.

To develop and deploy a cloud-based chatbot application, Traveloka Singapore can leverage the benefits of cloud computing technology (Wijaya and Wjiaya 2018). Cloud computing provides scalable and flexible infrastructure, enabling the chatbot application to run optimally and handle varying levels of user traffic. Cloud computing also ensures high availability and reliability of the chatbot application, minimizing downtime and enhancing user experience.

In conclusion, integrating cloud-based chatbot technology in customer service operations can enhance the quality of customer service Traveloka Singapore provides. This innovative solution can reduce response time, automate processes, and improve user satisfaction, contributing to the platform's overall success.

MATERIALS AND METHODS

1. Data Collection Technique

a. Observation

To get datasets related to this chatbot, developers look for datasets on many dataset provider websites such as Kaggle. The developer also uses frequently asked questions on the Traveloka Singapore website as an additional dataset for the chatbot application.

b. Interview

The developer conducted an online interview with Mr. Juan Kanggrawan as PIC of Traveloka Singapore via WhatsApp message by asking several things related to the design of the program to be developed.

c. Literature Review

The developer conducted a literature study by looking for books and journals related to the design of this chatbot application.

2. System Development Model

The developer designs this chatbot application using the agile method with the Scrum framework. According to Schwaber & Sutherland (Hadji, Taufik, and Mulyono 2019), Scrum is a framework that can address complex and ever-changing problems and provide suitable product quality according to user needs creatively and productively. The scrum method has several steps that need to be done, such as

a. Product Backlog

The product backlog in developing this chatbot application is the development of machine learning models, deploying machine learning models using APIs, and displaying them in the Android mobile application.

b. Sprint Backlog

The sprint backlog in the development of this application will later divide into several more detailed phases of the product backlog that have been defined in the product backlog process.

c. Sprint Planning

Sprint planning in developing this application will perform system design and analysis using UML diagrams.

d. Sprint

Sprint itself is the heart of the Scrum method, which consists of several work plans that have been previously divided. Sprints are defined in the backlog that must be completed within a pre-agreed time.

e. Sprint Review dan Sprint Retrospective

After the sprint, the application must be reviewed to inspect and adapt the product backlog as needed. Next, make a retrospective sprint so that the application that has been developed gets input, whether it is by the requirements or not. It will be readjusted and added to the backlog if it does not match. If appropriate, then the application will be tested.

RESULT AND DISCUSSION

1. Product Backlog

The product backlog in developing this chatbot application is the development of machine learning models, deploying machine learning models using APIs, and displaying them in the Android mobile application.

Table 1. Product Backlog

No	Product Backlog
1	Dataset gathering
2	Model machine learning development
3	Training dan evaluation model
4	Export model machine learning
5	API endpoint development
6	API testing development
7	Prepare infrastructure server
8	API deployment
9	API testing production
10	UI/UX apps
11	Implement UI/UX apps
12	Apps testing development
13	API integration
14	Apps testing production

Fourteen activities will be divided into three learning paths: Machine Learning, Cloud Computing, and Mobile Development.

2. Sprint Backlog

The sprint backlog in the development of this application will later divide into several more detailed phases of the product backlog that have been defined in the product backlog process.

Table 2. Sprint Backlog

No	Backlog Name	Priority	Task	Estimated (Hours)
1	Dataset gathering	100	Defines the dataset to be used Searching for datasets on Kaggle or other platforms	4 4
2	Model machine learning development	100	Develop models from existing datasets	40
3	Training dan evaluation model	100	Conduct training and evaluation of machine learning models	4
4	Export model machine learning	100	Export the finished model	2
5	API endpoint development	100	Looking for references for developing APIs with Flask API Develop API endpoints based on research results	4 40
6	API testing development	100	Doing API testing after development	2
7	Prepare infrastructure server	100	Creating server infrastructure on the Google Cloud Platform	4
8	API deployment	100	Deploy API endpoints on the server	4
9	API testing production	100	Perform API testing after deployment	2
10	UI/UX apps	100	Create Android application design	4
11	Implement UI/UX apps	100	Creating Android application pages	40
12	Apps testing development	100	Perform functional testing of the application after the page is created	4
13	API integration	100	Perform API endpoint integration within the application	16
14	Apps testing production	100	Perform functional testing of applications after integration	4

The table above shows the division of tasks and responsibilities to be carried out by each member involved, along with the allocation of time required to perform those tasks and responsibilities.

3. Sprint Planning

Sprint planning in developing this application will perform system design and analysis using UML diagrams.

a. Diagram Activity

PILAR Nusa Mandiri: Journal of Computing and Information System
Vol. 19, No. 1 March 2023 | DOI: 10.33480/pilar.v1i1.3055

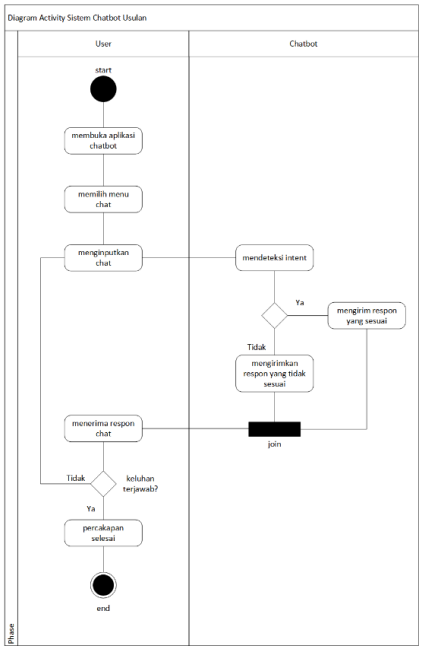

Figure 1. Diagram Activity

The activity diagram above will occur when the user uses the application.

b. Apps Infrastructure

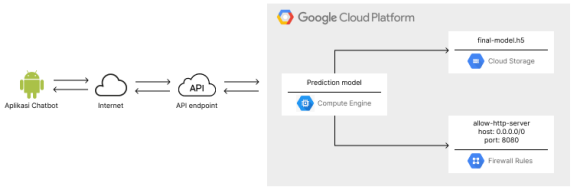

Figure 2. Apps Infrastructure

The developed application is Android-based and connected to an API deployed using the Google Cloud Platform through VM Instances. The machine learning model is located in Cloud Storage, and the firewall is configured to enable the API to be accessed by the application.

4. Sprint

Sprint itself is the heart of the Scrum method which consists of several work plans that have been previously divided. Sprints are defined in the backlog that must be completed within a pre-agreed time. In developing this chatbot application, it is divided into 3 sprints consisting of several tasks for each team.

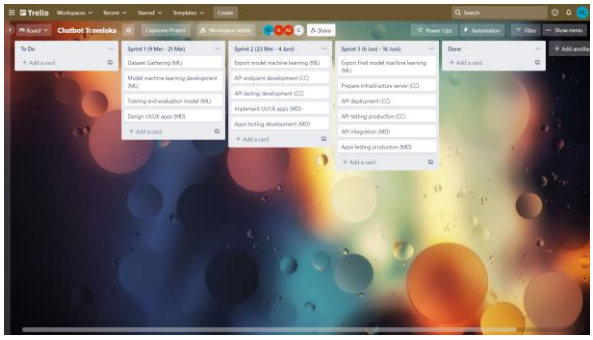

Figure 3. Sprints on Trello

The above is a visualization of the Trello application showing the task allocation of each learning path.

5. Sprint Review dan Retrospective

a. Sprint Review

After the sprint, the application must be reviewed to inspect and adapt the product backlog as needed.

1) Sprint 1 (9 May – 21 May)

a) Dataset Gathering

Figure 4. Dataset Sample

The above is an example of a dataset the machine learning team obtained.

b) Model machine learning development

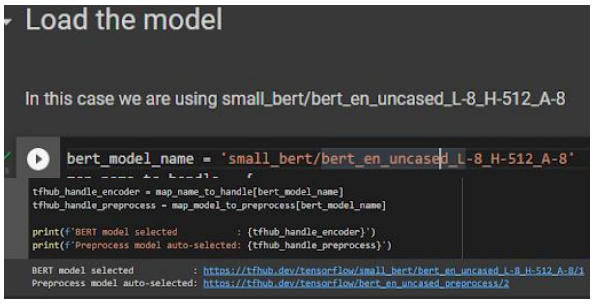

Figure 5. Sample Model Machine Learning

The above is a code snippet used in developing the machine learning model.

c) Training and evaluation model machine learning

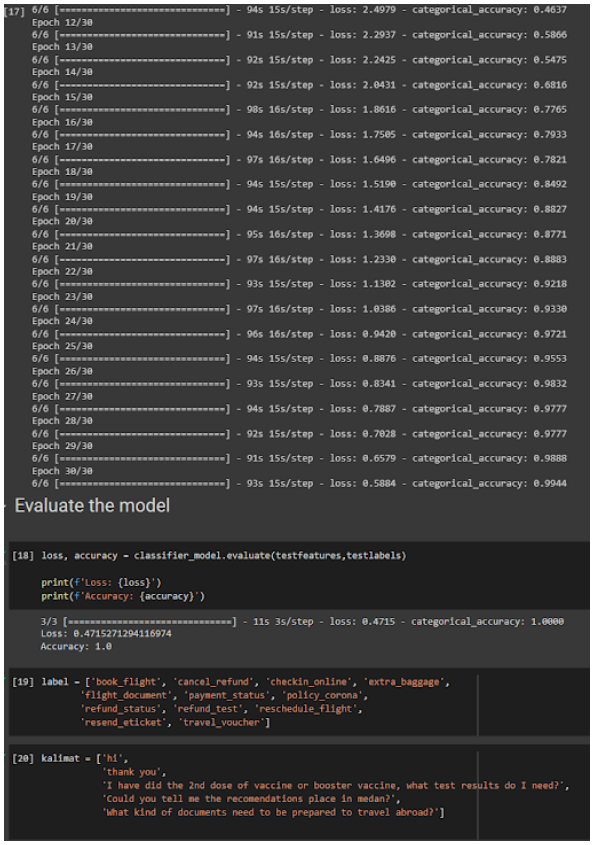

Figure 6. Result of Training and Evaluation Model

The above Figure 6 shows the results of the training and evaluation.

d) Design UI/UX Apps

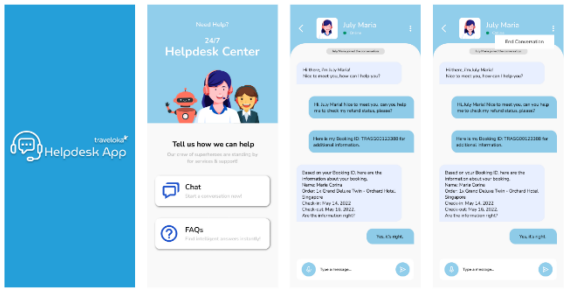

Figure 7. Apps UI/UX Design

In application development, it is incomplete without any design to be developed. The above is an example of a design to be developed.

2) Sprint 2 (23 May - 4 June)

a) Export model machine learning

Figure 8. Initial Model Machine Learning

The machine learning model that is ready to use will then be exported to Google Cloud Storage. In Figure 8 above, the model was first uploaded to Google Drive.

b) API Endpoint Development

Figure 9. Example of Script API Endpoint

Figure 9 shows a snippet of the API code being developed so that the machine-learning model can be used in the future.

c) API Testing Development

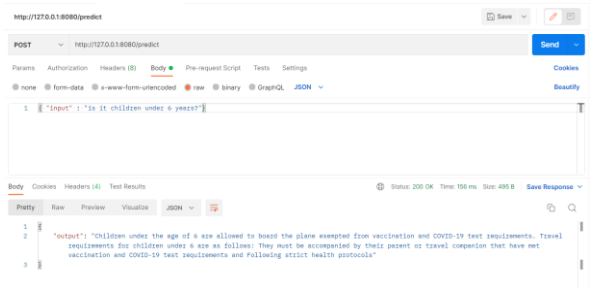

Figure 10. API Endpoint Development Result Testing

After developing the previously conducted API, it is necessary to conduct testing to ensure that the API and machine learning model can function correctly.

d) Implement and Testing UI/UX Apps

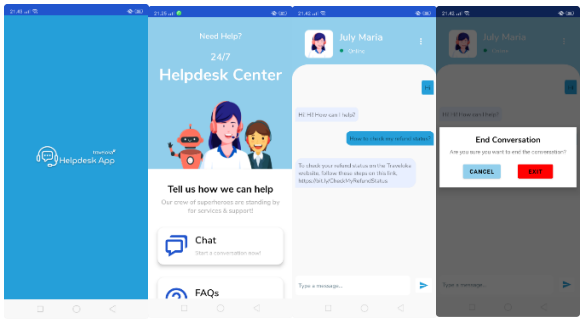

Figure 11. Implement and Testing UI/UX Apps

PILAR Nusa Mandiri: Journal of Computing and Information System
Vol. 19, No. 1 March 2023 | DOI: 10.33480/pilar.v1i1.3055

From the design presented in Figure 7 above, it will be developed into an interactive application, as shown in Figure 11.

3) Sprint 3 (6 June – 16 June)

a) Final Model Machine Learning

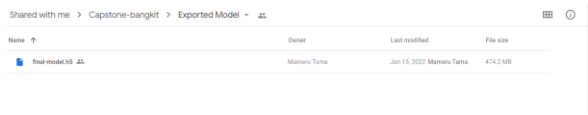

Figure 12. Final Model Machine Learning

In system development, there will inevitably be trial and error. Figure 12 shows the final machine learning model the Machine Learning team has improved.

b) Prepare Infrastructure Server

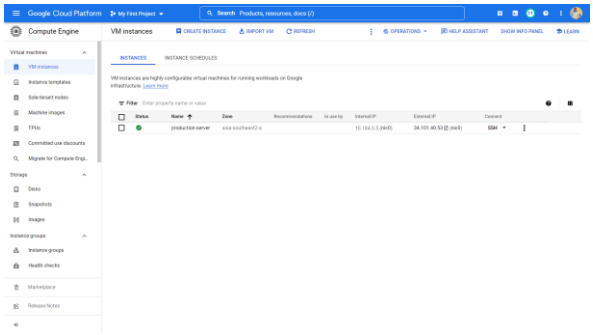

Figure 13. Production Server with Compute Engine

The Cloud Computing team will prepare a server that will be used for deployment, as shown in Figure 13, using Compute Engine.

c) API Deployment

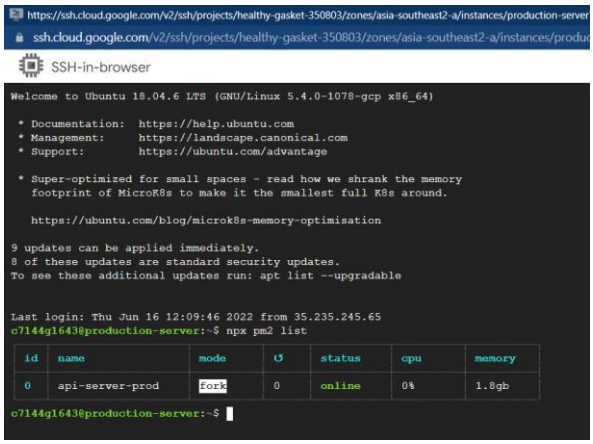

Figure 14. Deployed API Server

Figure 14 shows that the server is ready to use and deployed successfully. The API server will remain active using Process Manager (PM2).

d) API Testing Production

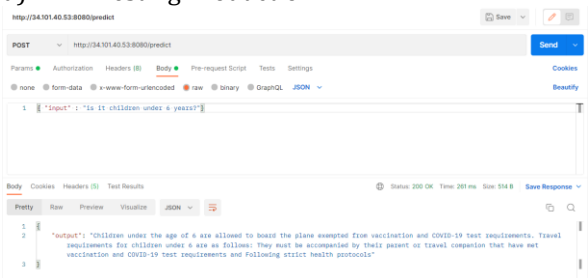

Figure 15. API Testing Production

Before the Mobile Development team uses the API for integration with the application, the API must be tested first so that the results obtained can meet expectations.

e) API Integration dan Testing Production

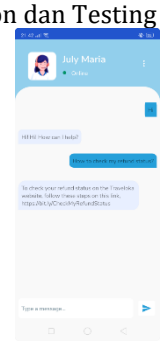

Figure 16. API Integration and Testing Production

The Mobile Development team will integrate the deployed API into the application and perform retesting to ensure the results match the desired outcome.

b. Sprint Retrospective

Next, make a retrospective sprint so that the application that has been developed gets input, whether it is by the requirements or not. It will be readjusted and added to the backlog if it does not match. If appropriate, then the application will be tested.

CONCLUSION

Based on the above discussion regarding the development of cloud computing-based chatbot applications, the author can conclude that this chatbot application was developed based on cloud computing, which means the entire process of machine learning models and APIs was developed entirely using cloud technology. With this application, it is hoped that it will make it easier for users to make complaints and can reduce repeated questions that users often ask. The development of this chatbot application uses the agile method with the scrum framework. Where in the development process carried out is divided into several phases

called sprints. The development of this application is carried out in 3 sprints starting from the first time this project was announced to completion.

REFERENCE

Ananda Dwi, Firdha Imamah, Yusuf Mei Andre, and Ardiansyah. 2018. "Aplikasi Chatbot (Milki Bot) Yang Terintegrasi Dengan Web CMS Untuk Customer Service Pada UKM MINSU." *Jurnal Cendikia* XVI:100–106.

Astuti, Rani Natadian and Muhamad Fatchan. 2019. "Perancangan Aplikasi Teknologi Chatbot Untuk Industri Komersial 4.0." *Prosiding Seminar Nasional Teknologi Dan Sains (SNasTekS)* 0(September):339–48.

Çallı, Levent and Fatih Çallı. 2022. "Understanding Airline Passengers during Covid-19 Outbreak to Improve Service Quality: Topic Modeling Approach to Complaints with Latent Dirichlet Allocation Algorithm." *Transportation Research Record: Journal of the Transportation Research Board* 036119812211120.

Chandra, Yudi Irawan and Kosdiana. 2019. "Rancang Bangun Aplikasi Chat Bot Line Menggunakan Pendekatan Agile Process Dengan Model Extreme Programming Berbasis Web (Studi Kasus Di STMIK JAKARTA STI&K)." *Seminar Nasional Teknologi Informasi Dan Komunikasi STI&K (SeNTIK)* 3(1):149–60.

Enterprise, Jubilee. 2018. *Teknik Memasang Chatbot Di Toko Online*. Yogyakarta: Elex Media Komputindo.

Hadji, Shon, M. Taufik, and Sri Mulyono. 2019. "Implementasi Metode Scrum Pada Pengembangan Aplikasi Delivery Order Berbasis Website (Studi Kasus Pada Rumah Makan Lombok Idjo Semarang)." *Konferensi Ilmiah Mahasiswa Unissula (KIMU)* 32–43.

Larasati Amalia, Eka and Dimas Wahyu Wibowo. 2019. "Rancang Bangun Chatbot Untuk Meningkatkan Performa Bisnis." *Jurnal Ilmiah Teknologi Informasi Asia* 13(2):137–42.

Li, Chia Ying, and Jin Ting Zhang. 2023. "Chatbots or Me? Consumers' Switching between Human Agents and Conversational Agents." *Journal of Retailing and Consumer Services* 72:103264.

Nugroho, Adi, Derry Pramono Adi, and Agustinus Bimo Gumelar. 2020. "Chatbot Untuk Customer Service Berbasis Teks Dan Suara Pada Sistem Manajemen Pemesanan (OMS) Menggunakan Platform Android." *Jurnal Repositor* 2(6):683.

Sudaryono, Sudaryono, Nuke Puji Lestari, and Ketut Gunawan. 2020. "Perancangan Virtual Assistant Entrepreneurship Menggunakan Metode Scrum." *Journal of Innovation And Future Technology (IFTECH)* 2(2):66–77.

Taylor, Charles R., Philip J. Kitchen, Matthew E. Sarkees, and Christian O. Lolk. 2020. "Addressing the Janus Face of Customer Service: A Typology of New Age Service Failures." *European Journal of Marketing* 54(10):2295–2316.

Wijaya, Gita Surja and Irianto Wjiaya. 2018. *Bedah Total Server: Referensi Lengkap Teknologi Server, Data Center, Virtualization, Cloud Computing & Enterprise System*. Jakarta: Gramedia Pustaka Utama.

PILAR Nusa Mandiri: Journal of Computing and Information System
Vol. 19, No. 1 March 2023 | DOI: 10.33480/pilar.v1i1.3055