



INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv



Chatbot for Diagnosis of Pregnancy Disorders using Artificial Intelligence Markup Language (AIML)

Alam Rahmatulloh^a, Anjar Ginanjar^a, Irfan Darmawan^{b,*}, Neng Ika Kurniati^a, Erna Haerani^a

^a Department of Informatics, Faculty of Engineering, Siliwangi University, Tasikmalaya, Indonesia

^b Department of Information System, Faculty of Industrial Engineering, Telkom University, Bandung, Indonesia

Corresponding author: *irfandarmawan@telkomuniversity.ac.id

Abstract—Artificial Intelligence has evolved in sophistication and widespread use. This study aims to create a chatbot application in the health sector regarding the early diagnosis of pregnancy disorders. Based on basic health research, only 44 percent of pregnant women know the danger signs of pregnancy. The chatbot application developed is expected to facilitate and increase knowledge for pregnant women about the danger signs of pregnancy, especially early diagnosis of pregnancy disorders. The chatbot application was developed with artificial intelligence technology based on Artificial Intelligence Markup Language with the question-answer concept using the Pandorabots framework. The test is carried out in two stages: functional and pattern matching. The functional testing uses the black-box testing method, and the pattern-matching test on the chatbot uses the sentence similarity and bigram methods based on user input and keywords similarity in the bot's knowledge base. The functional testing results show that the chatbot application runs well, with the eligibility criteria reaching 81.4% and the results of the keyword similarity test (pattern matching) are zero to one, in the sense that the value of one has the same similarity between user input and pattern. Meanwhile, the zero value has no similarities, so the bot will respond to it as free input. So it can be concluded that the bot can respond to user questions when the pattern and input have the same level of similarity.

Keywords—AIML; artificial intelligence; chatbot; diagnosis.

Manuscript received 15 Apr. 2022; revised 9 Aug. 2022; accepted 12 Dec. 2022. Date of publication 31 Mar. 2023.
International Journal on Informatics Visualization is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.



I. INTRODUCTION

The maternal mortality rate in Indonesia is still at an alarming level. For example, reviewing data from the Indonesian Demographic and Health Survey (IDHS), 359 deaths out of 100,000 births in 2012 [1]. Furthermore, the death rate is still relatively high until the end of 2019, which is as many as 305 cases out of 100,000 births [2]. The problem that often occurs today is the lack of knowledge of pregnant women about the symptoms felt during pregnancy (antenatal). This problem makes pregnant women unaware of the dangers of specific symptoms that trigger dangerous diseases and lead to miscarriage and even death.

Based on data from the United Nations Children's Fund (UNICEF), the average pregnant woman made her first pregnancy visit in Indonesia in 2012 which was 72%, and this did not meet the rules recommended by the ministry of health, which is at least four pregnancy visits [3]. These data are relevant to the Basic Health Research (Riskesdas) results related to the percentage of knowledge of pregnant women

about pregnancy disorders, where only 44% of mothers know about the dangers of pregnancy disorders. Factors that influence the lack of knowledge of pregnant women include education, environmental culture, and information [4]. So the importance of knowledge about the dangerous signs of pregnancy, including early diagnosis of pregnancy disorders for pregnant women, is vital.

Overcoming the lack of knowledge for pregnant women about the danger signs of pregnancy, especially the problem of pregnancy disorders, the proposed solution is to create a system using artificial intelligence (AI) technology and then implement it into a program in the form of a chatbot. Chatbots use AI techniques such as machine learning (ML) and natural language processing (NLP) [5]–[8]. Today, chatbots are integrated into every business, including education [7], [9], banking [10], transportation [11], communication [12], [13], hospitals [14]–[16], and other services [17]. Likewise, in e-commerce sales and marketing [18], an intelligent knowledge-based conversational agent system to support customer service has been implemented [19]. Therefore, the

implementation of the chatbot program in dealing with this problem is considered appropriate because the bot has an interactive program and can interact with users in the form of responses declared in the database [20].

The first chatbots were ELIZA and ALICE (Artificial Linguistic Internet Computer Entity). ELIZA is processed using computational linguistics with simple parsing and substituting keywords into reframed phrases [21]. Meanwhile, ALICE can convert readable text into artificial intelligence markup language (AIML) to reveal possible functional prototypes without complex ML algorithms [22].

Artificial Intelligence (AI) technology for chatbots has grown in sophistication and widespread use, one of which is Artificial Intelligence Markup Language (AIML). AIML is derived from Extensible Markup Language (XML), used to build up conversational agents artificially [23]. Artificial Intelligence Markup Language (AIML) applies the concept of pattern matching, namely by matching input from the user with a predetermined pattern of rules. The pattern-matching process in AIML is knowledge of pattern rules used to select responses related to one another [24]. The application of AIML can also be combined with emotional models to increase trust [25].

The chatbot development in this study uses Artificial Intelligence Markup Language (AIML) with the Question Answer System method [26]. Question Answer is a system that applies the concept of natural language to express information in phrases as a response to an answer in a more specific and natural form [27]. The pattern-matching testing process uses the sentence similarity and bigram methods between user input and the existing pattern on the bot's knowledge base, and functional testing is carried out using the black box testing method.

II. MATERIAL AND METHOD

A. Artificial Intelligence Markup Language (AIML)

Artificial Intelligence Markup Language (AIML) is a complementary language to Extensible Markup Language (XML) that allows users to create or customize a Bot's knowledge base. Artificial Intelligence Markup Language (AIML) was developed by the Alice bot community and has become the basis of the first Alice bot, the Artificial Linguistic Internet Computer Entity (ALICE).

B. Pandorabots

Pandorabots is an open-source framework that can develop and publish on the web and mobile. Pandorabots uses Artificial Intelligence Markup Language (AIML) as its knowledge content markup language. "Pandorabots is the largest community of chatbots on the internet." As of February 2012, the free community service Pandorabots is home to more than 166,000 botmasters and 201,000 chatbots in multiple languages [28].

C. Related Works

Previous research on the implementation of chatbots with Artificial Intelligence (AI) technology using Artificial Intelligence Markup language (AIML) has been widely carried out, including by Rusmarasy, Priyambadha, and

Pradana [29], which is used as a motivator in the E-learning Code Maniac website, which aims to motivate website users.

Other research has been carried out by Bahartyan, Bahtiar, and Waspada [28], as a virtual assistant that functions as a center for ordering and searching for products on the Eduando.com online bookstore website. The chatbot is Web-based with bot knowledge using Artificial Intelligence Markup Language (AIML). System development method using ripple with a functional testing system with a black box.

Many studies related to chatbots use AIML as their knowledge base with different concepts and case studies. This study focuses on implementing chatbots as information media to diagnose early symptoms of pregnancy disorders with the concept of mobile-based question answers using the Pandorabots framework. The test will be carried out in two ways: the symptom pattern matching test and the user acceptance test.

D. Research Method

The methodology used in this research is a qualitative method with several stages, including data collection, designing a chatbot application with a personal extreme programming (XP) development model, evaluation, and conclusions. The design of the chatbot application for diagnosing pregnancy disorders uses the Personal extreme programming (XP) development model, which includes seven stages, including requirements, planning, iteration initialization, design, implementation, system testing, and retrospective. This model tends to use an object-oriented approach.

III. RESULT AND DISCUSSION

A. Requirements

This stage analyzes the system requirements needed for developing a chatbot application for diagnosing symptoms of pregnancy disorders, including symptom data, data on pregnancy disorders accompanied by solutions, and functional analysis that will be implemented in the chatbot application along with hardware and software specifications.

B. Planning

The system requirements used in developing chatbot applications include symptom and disease data, solutions, functional analysis, and hardware and software. Table 1 shows a sample of data on symptoms and diseases of pregnancy disorders, and table 2 shows a solution from the sample data for these symptoms and diseases.

TABLE I
DATA SYMPTOMS AND DISEASES OF PREGNANCY

| Symptoms | Anemia | Typhus | Eclampsia |
|---------------------|--------|--------|-----------|
| Dizzy | √ | | √ |
| Pale | √ | | |
| Tired easily | √ | | |
| High fever | | √ | |
| Diarrhea | | √ | |
| Stomachache | | √ | |
| Nausea and Vomiting | | | √ |
| Severe Headache | | | √ |
| High blood pressure | | | √ |

TABLE II
DISEASE DATA AND SOLUTIONS

| Disease Name | Solutions |
|--------------|---|
| Anemia | Carry out regular check-ups with the obstetrician to check the hemoglobin (Hb) of the blood. |
| Typhus | Follow the vaccination program provided by the government, and maintain food hygiene. |
| Eclampsia | Carry out regular check-ups with the doctor to monitor the condition of the mother and fetus; it is recommended to sleep on your left side. |

1) *Functional*: The functional analysis needed in the development of chatbot applications includes:

- Applications can display chat services with bots
- Applications can receive input questions from users.
- Applications can provide answers related to user questions.
- The application can display information about signs of pregnancy.
- The application can display a good sleeping position, good nutrition, and poor nutrition during pregnancy.

2) *Hardware dan Software*: Hardware and software specifications for making chatbot applications for the early diagnosis of pregnancy disorders are listed in Table 3.

TABLE III
HARDWARE AND SOFTWARE

| Hardware/Software | Minimum Spec |
|----------------------|-------------------------|
| Operating System | Windows 7 |
| RAM PC | 2 Gb |
| Storage | 250 Gb |
| Processor PC | Intel inside |
| Mobile OS | Android 6 (Marshmallow) |
| RAM SmartPhone | 512 Mb |
| Storage (internal) | 8 Gb |
| Processor Smartphone | Qualcomm Snapdragon |

C. Iteration Initialization

Iteration Initialization is the stage of designing the system functionality of the chatbot application for diagnosing symptoms of pregnancy disorders, including use case diagrams. The use case diagram of the chatbot application is shown in Figure 1.

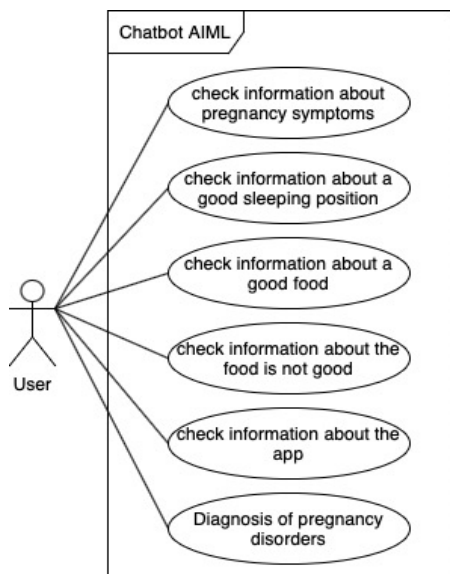


Fig. 1 Use Case Diagram

D. Design

This stage includes designing the interface of the chatbot application for the diagnosis of pregnancy disorders based on Android. An example of the main menu display of the user interface is shown in Figure 2.

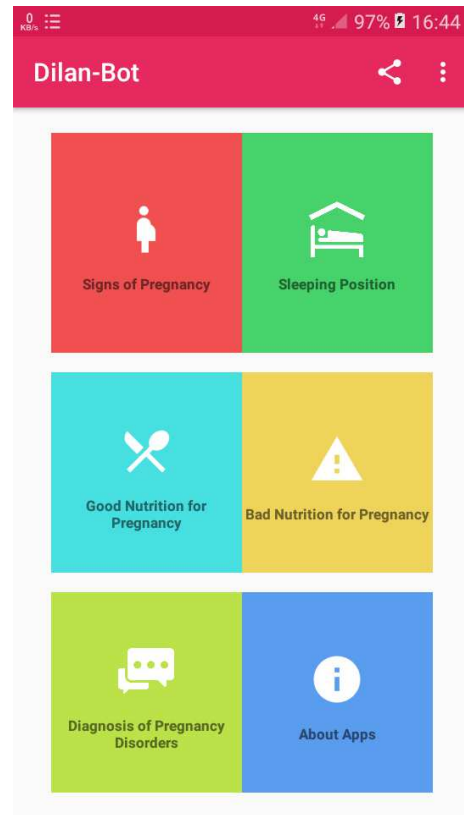


Fig. 2 Main menu display

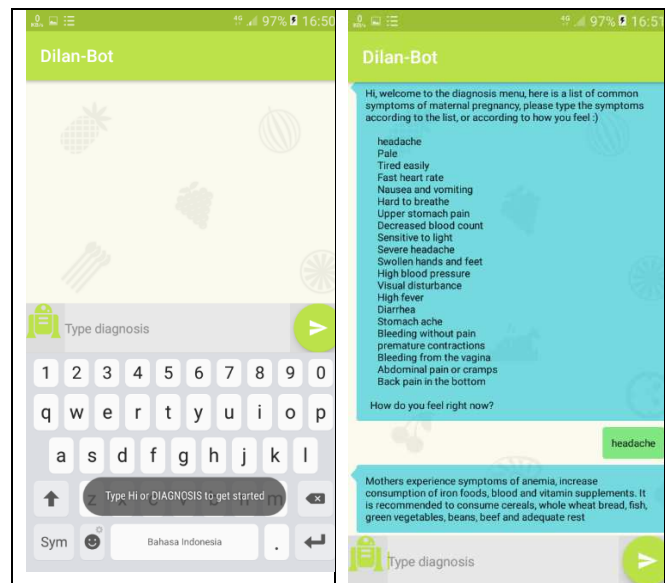


Fig. 3 Chatbot

E. Implementation

The implementation phase includes coding and the results of the chatbot application display for diagnosing pregnancy disorders, along with some knowledge base code snippets as follows:

```

<?XML version="1.0" encoding="UTF-8"?>
<aiml version="2.0">
<!-- insert your AIML categories here -->
<category>
<pattern>diagnose</pattern> <template>Hi, welcome to the
CiBumil chat service, a digital assistant who is ready
to help diagnose disorders of maternal pregnancy
symptoms :) The following are common symptoms of
maternal pregnancy disease:
Dizzy
Pale
Tired easily
Faster and irregular heartbeat
Severe headache
Hard to breathe
Nausea and vomiting
Upper abdominal pain
Sensitive to light
Decrease the number of blood platelets
High blood pressure
Swollen hands and feet
Visual disturbance
High fever
Diarrhea
Stomach ache
Painless bleeding in the second or third pregnancy
premature contractions
Bleeding from the vagina during early pregnancy
Abdominal pain or cramps
Back pain at the bottom

<!-- anemia disease-->
<category>
<pattern>dizzy</pattern>
<template>Mother has symptoms of anemia, namely a lack
of red blood cells, and consumes more iron, such as
vegetables and meat.
</template></category>
<category>
<pattern>pale</pattern>
<template><srai>dizzy</srai></template></category>
<category>
<pattern>tired easily</pattern>
<template><srai>dizzy</srai></template></category>
<category>
<pattern>fast and irregular heartbeat</pattern>
<template><srai>dizziness</srai></template>
</category>
</aiml>

```

A snippet of AIML-based bot knowledge of the symptoms and diseases of pregnancy disorders above can be seen in the code, and there are several tags, including the category tag, which defines the unit of bot knowledge. The pattern tag defines a pattern or input questions related to the bot's knowledge. The template tag defines the response from the bot associated with the previous pattern tag. The template tag allows passing to another pattern tag with the <srai> element. <srai> is a form of implementation of the equation of meaning. For example, one pattern is used to reference other patterns in the code, namely the "dizzy" pattern. The dizziness pattern has a template for anemia symptoms. That way, when the user inputs dizziness symptoms, the bot will answer "anemia." Likewise, with the symptoms of paleness, fatigue, fast heart rate, nausea, and vomiting, the bot's response is still the same, namely "anemia" because it refers to the "dizzy" pattern. More details in Figure 3 show the logical pattern of the bot, which refers to a pattern using the <srai> element.

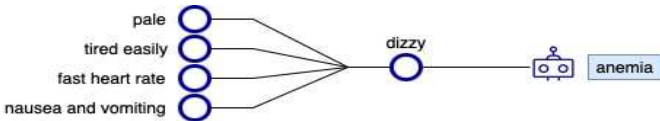


Fig. 4 One input pattern

Figure 4 explains that when a user diagnoses the chatbot application by only entering one symptom word, including dizziness, pale, quickly tired, fast heart rate, or nausea and vomiting, the bot responds by answering "anemia." To make it easier for users to diagnose more than one symptom in the same disease, condition tags can be used. The following code is listed in more detail.

```

<category>
<pattern>dizzy *</pattern>
<templates>
<think><setname="dizzy"> <star/></set></think>
<condition name="anemia">

<li value = "pale">Anemia</li>
<li value = "tired easily">Anemia</li>
<li value = "fast heartbeat"> Anemia</li>
<li value = "nausea and vomiting"> Anemia</li>
...
<li>Invalid input; maybe your input has a spelling error
or typo. Recheck your input and try again properly and
correctly according to the symptoms in the previous
diagnosis menu!</li>
</condition>
</templates>
</category>

```

Bot knowledge base code with the use of condition tags. The condition tag creates an IF-THEN-ELSE type in the bot response. The bot's response is conditioned on one disease, namely "anemia" in the code. This is done by checking the value of the predicate (anemia) in the list of elements (). If the value of the predicate matches, then the text of that element is called, and if the value of the predicate does not match the value in the list of elements (), then the element (which has no value attribute) will be called. The condition tag refers to the "dizzy" pattern tag followed by a * (wildcard), the wildcard sign is useful as free input, but in this case, it is only in the scope of the "anemia" predicate value. A more detailed description of the conditional tag logic pattern for the two inputs is shown in Figure 5.

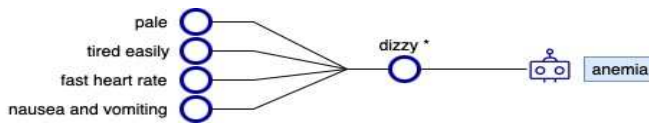


Fig. 5 Two input pattern

Figure 5 explains when a user diagnoses symptoms with two inputs referring to one keyword, dizziness followed by a * (wildcard) sign, and the bot responds by answering anemia. Using the condition tag, the concept is still the same for more than two inputs with different symptoms and diseases.

F. System Testing

This stage includes functional testing of the chatbot application system for diagnosing symptoms of pregnancy disorders using a black box while testing the pattern-matching bot using bigram and sentence similarity and testing the feasibility of the application using the User Acceptance Test (UAT) method.

1) Pattern matching testing: Pattern matching testing is done by matching keywords in the bot's knowledge base with user input related to these keywords. The pattern matching test uses the bigram and sentence similarity methods. In linguistics, a bigram means a pair of sequential written units

in letters, syllables, or words [24]. In another study, a bigram is a series of two consecutive sentences or letters [30]. An example of using a bigram, a two-letter pair of a word, is shown below.

```
Strings: "sick"
Bigram: {"si", "ic", "ck" }
```

While sentence similarity is a process to calculate the level of similarity of two sentences or strings [24], [30], the score generated from the similarity calculation is the value of zero and one. About bigrams. The simple formula to calculate it is listed in equation (1).

$$\frac{\text{count}(b1 \in b2) + \text{count}(b2 \in b1)}{\text{count}(b1) + \text{count}(b2)} \quad (1)$$

In equation (1), b1 is a bigram of the result of the first string, and b2 is the set of bigrams resulting from the second string, count(b1∈b2) as a set operation for each number of bigrams in b1, which is a member of the bigram in b2, and vice versa for count (b2∈b1). Count (b1) is the number of bigrams in the first string and vice versa for count (b2) [24].

Table 4 is an example of patterns and inputs used as pattern-matching tests. There is a "dizzy" string in the pattern section, while in the input section, there are four variations of the string, one of which resembles the pattern. The testing process is as follows.

```
String 1: "dizzy" (pattern)
String 2:"dizzy" (input)
b1 = {"di", "iz", "zz", "zy"}
b2 = {"di", "iz", "zz", "zy"}
b1 ∈ b2 = {"di", "iz", "zz", "zy"}
b2 ∈ b1 = {"di", "iz", "zz", "zy"}
maka diperoleh nilai string 1 dan nilai string 2
berdasarkan rumus sentence similarity menggunakan bigram
merujuk pada persamaan (1) yaitu:
b1 ∈ b2 = 4      b21 = 4
b2 ∈ b1 = 4      b2 = 4
= 8 / 8
= 1

String 1: "dizzy" (pattern)
String 2: "little dizzy" (input)
b1 = {"di", "iz", "zz", "zy"}
b2 = {"li", "it", "tt", "tl", "le", "di", "iz", "zz", "zy"}
b1 ∈ b2 = {"di", "iz", "zz", "zy"}
b2 ∈ b1 = {"di", "iz", "zz", "zy"}
then obtained
b1 ∈ b2 = 4      b1 = 4
b2 ∈ b1 = 4      b2 = 9
= 8 / 13
= 0,615

String 1: "dizzy" (pattern)
String 2: "very dizzy" (input)
b1 = {"di", "iz", "zz", "zy"}
b2 = {"ve", "er", "ry", "di", "iz", "zz", "zy"}
b1 ∈ b2 = {"di", "iz", "zz", "zy"}
b2 ∈ b1 = {"di", "iz", "zz", "zy"}
then obtained
b1 ∈ b2 = 4      b1 = 4
b2 ∈ b1 = 4      b2 = 7
= 8 / 11
= 0,727

String 1: "dizzy" (pattern)
String 2: "feel dizzy" (input)
b1 = {"di", "iz", "zz", "zy"}
b2 = {"fe", "ee", "el", "di", "iz", "zz", "zy"}
b1 ∈ b2 = {"di", "iz", "zz", "zy"}
b2 ∈ b1 = {"di", "iz", "zz", "zy"}
then obtained
b1 ∈ b2 = 4      b1 = 4
b2 ∈ b1 = 4      b2 = 7
= 8 / 11
= 0,727
```

The value of each pattern and input string is obtained, and the pattern and input values are calculated to determine the level of similarity. The final result is a similarity value with a range of values from zero to one. More details on the pattern and input testing results are listed in Table 5.

TABLE IV
PATTERN TEST DATA AND INPUT

| Pattern | Input User |
|---------|--------------|
| Dizzy | dizzy |
| | little dizzy |
| | very dizzy |
| | feel dizzy |

TABLE V
PATTERN AND INPUT TEST DATA RESULTS

| Pattern | Input | Similarity |
|---------|--------------|------------|
| Dizzy | dizzy | 1 |
| | Little dizzy | 0,615 |
| | Very dizzy | 0,727 |
| | Feel dizzy | 0,727 |

Based on the pattern and input test data results in Table 5, the highest similarity result is the "dizzy" input string with a value of 1, which is exactly resembling the "dizzy" pattern string. Likewise, the lowest value is found in the "little dizzy" input string, which has a value of 0.615. Therefore, based on the test results, it can be concluded that the bot can respond to user questions when the pattern and input have the same level of similarity. Unless the pattern has a unique tag, one of which is the * (wildcard) tag as an auxiliary word or free input, the bot responds by referring to one keyword from the pattern tag that has the wildcard tag.

2) *User acceptance test:* The data used for testing the User Acceptance Test (UAT) is based on the questionnaire results. The questionnaire consists of six questions, each having a weight value. As stated in equation (2), the Likert scale formula is used to calculate the percentage value of each answer to the questionnaire.

$$P = \frac{S}{\text{Ideal Score}} \times 100\% \quad (2)$$

In equation (2), P is the percentage value sought, and S is the number of frequencies multiplied by the total score of answers. The ideal score is the highest value times the number of samples. The percentage value of the answers to each question is shown in Table 6.

TABLE VI
PERCENTAGE OF QUESTIONS

| Question | Percentage |
|---|------------|
| Attractive application display | 82.5 % |
| The application menu is easy to use | 83.7% |
| The disease diagnosis feature with bots is easy to understand | 80% |
| Bots can provide accurate information about the diagnosis | 75.6% |
| Applications according to user needs, especially for pregnant women | 80% |
| Applications can be helpful for users, especially pregnant women | 86.9% |

After each value of the percentage of answers from each questionnaire is obtained with a Likert scale, the calculation

to find the overall percentage is based on the overall percentage formula shown in equation (3).

$$PK = \frac{\Sigma \text{Percentage Statement Sample}}{\Sigma \text{Overall percentage}} \quad (3)$$

$$PK = \frac{488.7\%}{600\%} \times 100\% = 81.4\%.$$

The overall percentage result is 81.4%. These results are obtained from all questionnaire calculations from each respondent's opinion. In addition, the chatbot application for diagnosing maternal pregnancy disorders is included in the eligibility criteria.

G. Retrospective

At the retrospective stage, they analyze the results of the progress of making a chatbot application to diagnose symptoms of pregnancy disorders to overcome errors from the application. The results of the analysis of the chatbot application for diagnosing the symptoms of pregnancy disorders can be concluded that the application is running well. Furthermore, the bot can provide disease information from the symptoms entered by the user. Therefore, the application is by the previous functional requirements.

IV. CONCLUSION

Based on the results of the research that has been carried out, it can be concluded that the implementation of Artificial Intelligence (AI) technology based on Artificial Intelligence Markup Language (AIML) on the chatbot system has been successfully applied to assist pregnant women in knowing the danger signs of pregnancy, including early diagnosis of pregnancy disorders. Furthermore, bots can respond to user input regarding the diagnosis of symptoms of pregnancy disorders and outside the topic of diagnosing pregnancy disorders. Based on the User Acceptance Test (UAT), the application functions well, with the feasibility of the application reaching 81.4%. Bots can be maximized by utilizing the tags contained in AIML to handle unknown user input. In addition, the knowledge of bots can be further developed, especially in the data section on symptoms and diseases of pregnancy disorders, so that bots can be more interactive in responding to user input.

REFERENCES

- [1] B. Kependudukan and B. Nasional, "Survei Demografi dan Kesehatan Indonesia," 2013.
- [2] S. Susiana, "Angka Kematian Ibu: Faktor Penyebab dan Upaya Penanganannya," *INFO Singkat: Kajian Singkat Terhadap Isu Aktual dan Strategis*, vol. XI, no. 24, pp. 13–18, 2015.
- [3] K. Ibu, "Kesehatan Ibu & Anak," no. Gambar 2, 2012.
- [4] B. Penelitian, D. A. N. Pengembangan, and K. K. Ri, "Riset kesehatan dasar," 2010.
- [5] R. Singh, M. Paste, N. Shinde, H. Patel, and N. Mishra, "Chatbot using TensorFlow for small Businesses," in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, Apr. 2018, pp. 1614–1619. doi: 10.1109/ICICCT.2018.8472998.
- [6] A. Følstad and P. B. Brandtzaeg, "Users' experiences with chatbots: findings from a questionnaire study," *Qual User Exp*, vol. 5, no. 1, p. 3, Dec. 2020, doi: 10.1007/s41233-020-00033-2.
- [7] D. Lee and S. Yeo, "Developing an AI-based chatbot for practicing responsive teaching in mathematics," *Comput Educ*, vol. 191, p. 104646, Dec. 2022, doi: 10.1016/j.compedu.2022.104646.
- [8] A. Rahmatulloh and H. Suhendy, "MikrobatX: Deep Learning Approach for Microscopic Identification and Classification of Medicinal Leaf Simplicia Fragments Using Sift Feature Extraction," *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.4226649.
- [9] P. Smutny and P. Schreiberova, "Chatbots for learning: A review of educational chatbots for the Facebook Messenger," *Comput Educ*, vol. 151, p. 103862, Jul. 2020, doi: 10.1016/j.compedu.2020.103862.
- [10] E. Mogaji, J. Balakrishnan, A. C. Nwoba, and N. P. Nguyen, "Emerging-market consumers' interactions with banking chatbots," *Telematics and Informatics*, vol. 65, p. 101711, Dec. 2021, doi: 10.1016/j.tele.2021.101711.
- [11] N. Thi Khanh Chi, "Transforming travel motivation into an intention to pay for nature conservation in national parks: The role of Chatbot e-services," *J Nat Conserv*, vol. 68, p. 126226, Aug. 2022, doi: 10.1016/j.jnc.2022.126226.
- [12] D. Ireland *et al.*, "Introducing Edna: A trainee chatbot designed to support communication about additional (secondary) genomic findings," *Patient Educ Couns*, vol. 104, no. 4, pp. 739–749, Apr. 2021, doi: 10.1016/j.pec.2020.11.007.
- [13] H. Jiang, Y. Cheng, J. Yang, and S. Gao, "AI-powered chatbot communication with customers: Dialogic interactions, satisfaction, engagement, and customer behavior," *Comput Human Behav*, vol. 134, p. 107329, Sep. 2022, doi: 10.1016/j.chb.2022.107329.
- [14] F. Jiang *et al.*, "Artificial intelligence in healthcare: past, present and future," *Stroke Vasc Neurol*, vol. 2, no. 4, pp. 230–243, Dec. 2017, doi: 10.1136/svn-2017-000101.
- [15] S. Siddique and J. C. L. Chow, "Machine Learning in Healthcare Communication," *Encyclopedia*, vol. 1, no. 1, pp. 220–239, Feb. 2021, doi: 10.3390/encyclopedia1010021.
- [16] P. Butow and E. Hoque, "Using artificial intelligence to analyse and teach communication in healthcare," *The Breast*, vol. 50, pp. 49–55, Apr. 2020, doi: 10.1016/j.breast.2020.01.008.
- [17] S. Colabianchi, M. Bernabei, and F. Costantino, "Chatbot for training and assisting operators in inspecting containers in seaports," *Transportation Research Procedia*, vol. 64, pp. 6–13, 2022, doi: 10.1016/j.trpro.2022.09.002.
- [18] C. V. Misischia, F. Poecze, and C. Strauss, "Chatbots in customer service: Their relevance and impact on service quality," *Procedia Comput Sci*, vol. 201, pp. 421–428, 2022, doi: 10.1016/j.procs.2022.03.055.
- [19] E. W. T. Ngai, M. C. M. Lee, M. Luo, P. S. L. Chan, and T. Liang, "An intelligent knowledge-based chatbot for customer service," *Electron Commer Res Appl*, vol. 50, p. 101098, Nov. 2021, doi: 10.1016/j.elerap.2021.101098.
- [20] D. S. Hormansyah and Y. P. Utama, "Aplikasi Chatbot Berbasis Web Pada Sistem Informasi Layanan Publik Kesehatan Di Malang Dengan Menggunakan Metode Tf-Idf," *Jurnal Informatika Polinema*, vol. 4, no. 3, p. 224, 2018, doi: 10.33795/jip.v4i3.211.
- [21] J. Weizenbaum, "ELIZA — a computer program for the study of natural language communication between man and machine," *Commun ACM*, vol. 26, no. 1, pp. 23–28, Jan. 1983, doi: 10.1145/357980.357991.
- [22] B. AbuShawar and E. Atwell, "ALICE Chatbot: Trials and Outputs," *Computación y Sistemas*, vol. 19, no. 4, Dec. 2015, doi: 10.13053/cys-19-4-2326.
- [23] Md. S. Satu, Md. H. Parvez, and Shamim-Al-Mamun, "Review of integrated applications with AIML based chatbot," in *2015 International Conference on Computer and Information Engineering (ICCIE)*, Nov. 2015, pp. 87–90. doi: 10.1109/ICCIE.2015.7399324.
- [24] A. Dewi and B. Setiaji, "Pemanfaatan Sentence-Similarity Measurement Untuk Proses Pencarian Pola Pada Chatbot Berbasis Pattern-Matching," *Seminar Nasional Teknologi Informasi dan Multimedia 2014*, pp. 39–44, 2014.
- [25] R. Sutoyo, A. Chowanda, A. Kurniati, and R. Wongso, "Designing an Emotionally Realistic Chatbot Framework to Enhance Its Believability with AIML and Information States," *Procedia Comput Sci*, vol. 157, pp. 621–628, 2019, doi: 10.1016/j.procs.2019.08.226.
- [26] Y. Sharma and S. Gupta, "Deep Learning Approaches for Question Answering System," *Procedia Comput Sci*, vol. 132, pp. 785–794, 2018, doi: 10.1016/j.procs.2018.05.090.
- [27] F. Azwary, F. Indriani, and D. T. Nugrahadi, "Question Answering System Berbasis Artificial Intelligence Markup Language," *Kumpulan Jurnal Ilmu Komputer*, vol. 04, no. 01, pp. 48–60, 2016.
- [28] E. Bahartyan, N. Bahtiar, and I. Waspada, "Integrasi Chatbot Berbasis Aiml Pada Website E-Commerce Sebagai Virtual Assistant Dalam Pencarian Dan Pemesanan Produk (Studi Kasus Toko Buku Online

- Edu4Indo.Com),” *Jurnal Masyarakat Informatika*, vol. 5, no. 10, 2015, doi: 10.14710/jmasif.5.10.34-43.
- [29] B. Rusmarasy, B. Priyambadha, and F. Pradana, “Pengembangan Chat Bot pada CoMa untuk Memberikan Motivasi Kepada Pengguna Menggunakan AIML,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 5, pp. 4484–4490, 2019.
- [30] J. Informatika, F. I. Komputer, and U. A. Yogyakarta, “Implementasi Algoritma Sentence Similarity Dicky Andhika Rizalldhi , 2 Galih Adhi Kuncoro Rosyad , 3 Anggit Dwi Hartanto,” vol. 4, no. 1, pp. 10–14, 2020.