psySSIST: A Mental Wellness-Oriented Conversational Assistant

Aniruddha Chakravarty
Dept. of Computer Science,
SOE, MIT-ADTU
Pune, Maharashtra, India

Kalpak Jibhakate
Dept. of Computer Science,
SOE, MIT-ADTU
Pune, Maharashtra, India

Nilima Kulkarni
Dept. of Computer Science,
SOE, MIT-ADTU
Pune, Maharashtra, India

Jyoti Malhotra
Dept. of Computer Science,
SOE, MIT-ADTU
Pune, Maharashtra, India

Abstract— **Programs that attempt to reproduce a human dialogue through an auditory/textual user interface to accomplish a specific purpose are known as conversational agents. Mental health issues such as depression, anxiety, etc., and their symptoms can prevent a person from performing daily tasks effectively. Digitization of mental health services using conversational agents will make it easier for them to have ubiquitous and accessible medication. In this manuscript, we present a conversational assistant whose aim is to engage the individual in a dialogue to help ease the effects of any underlying mental health problem. The suggested approach involves Natural Language Processing and Emotion Analysis. Our conversational assistant prototype is not designed for any medicinal or diagnostic purpose, nor is it intended as a replacement for appropriate medical and social assistance. It is meant to be the first point of contact seeking help between a patient (with a mental health condition or crisis) and an authorized physician.**

Keywords—Natural Language Processing (NLP), Neural Network, Sentiment Analysis, conversational response generation, Chatbot Systems, Chatbot Knowledge, Human Computer Interaction (HCI)

I. INTRODUCTION

Businesses have been motivated, in part due to the effects of the COVID-19 pandemic and the rise in cloud infrastructure, to deliver web-based services using user-accessible software. Almost every part of several business houses is now linked to the Internet, improving customer connectivity and problem-solving. Chatbots (also known as conversational assistants) are used in various areas of human-computer interaction (HCI) to support, add and substitute human activity [1].

A range of mental illnesses exist, which differ in terms of severity/impact on the individual patient. Some require patients to be treated urgently with clinical care whereas others require patients who do not require immediate psychological intervention [2]. Our conversational assistant aims to alleviate symptoms of mild mental disorders, allowing the end-user (patient) to have a better quality of life. This will enable the end-user to regain aspects of the life that were previously rendered inaccessible due to mental illness. In summary, we are building a chatbot that acts as a companion to individuals that tend to be on the depression spectrum.

We intend for individuals to easily access personalized mental counselling services by using a personalized conversational assistant (Chatbot). The Chatbot is called 'psySSIST' as the role of Chatbot is analogous to assisting individuals with psychological problems. A conversational assistant system that engages with users and asks them to explain their state using natural language was demonstrated.

In doing so, the conversational assistant is demonstrably capable of engaging in small talk with the end-user (patient) and performs inquires for relevant user data, e.g., name, age, etc., and appeals for mental health symptoms that the user faces. Based on user-inputted messages, the Dialogflow [3] based algorithm can retrieve patterns from messages that are relevant to the mental health symptoms that the user might be facing. The algorithm structure is such that specific questions are gradually asked to obtain a good diagnosis. The system was also compared to the existing Chatbot implementations. We illustrate that the proposed mental health Chatbot could be a better alternative to many existing medical science bots.

In the healthcare sector, the need for chatbots is highlighted to improve patient satisfaction. The three basic components of our system are:

- Accurately detecting symptoms and looking up documented symptoms in the database.
- Enabling symptom recognition and, if necessary, refer the patient to the most suitable specialist.
- Engaging end-user in conversation to determine additional data and diagnose underlying mental health condition.

The remainder of the article is structured as follows. The associated work is reviewed in Section II, which includes examining relevant precedents. Section III elaborates upon a survey that we performed to determine the end-user and their likely preferences. Section IV touches separately upon proposed system architecture and implementation, followed by Section V, which lists testing & results, along with some observations. Section VI gives the conclusion and future scope of this work and is followed by Section VII, which is composed of acknowledgments. Lastly, Section VIII is used to list the references.

II. RELATED WORK

This section depicts context of previous work done with regards to implementing functional Chatbot using a suitable technology stack. Area of work ranges from Chatbot deployment to Chatbot modelling frameworks along with Artificial Intelligence Markup Language (AIML), which is used to define general queries like "how do you do?" and "how can I help you?" in combination with Latent Semantic Analysis (LSA) within a Chatbot framework. Bhavika Ranoliya et al [4] came up with a Chatbot that utilized AIML and LSA to respond to FAQs related to a specific topic.

In their approach, a user gives a set of text-based inputs, which are used to reveal parallels between words as vector representation, allowing unanswered AIML queries are treated as a response by LSA. Subsequently, random responses can be generated for the same query. LSA can also be utilized to discover likeness between words in terms of vector representation [5]. Different types of context can be inferred from user input, ranging from determining user mood, and emotion.

Kyo-Joong Oh et al [6] stated that text, image, and video coupled with audio as an option can be used to build an emotion recognition algorithm. On building the Chatbot, they found that their emotional classification algorithm had 67.52% accuracy on average. Generalized Chatbot frameworks can be used to reduce the magnitude of problems faced by institutions.

Ho Thao Hien et al [7] believe in introducing generalized Chatbot frameworks, such that content delivery and process automation are sped up to enable minimal human intervention. Utilization of a Chatbot will help quicken up services, and existing resources (such as skilled humans/infrastructure) can be deployed elsewhere to boost the operational efficiency factor of the institution.

Keyword retrieval and determining their nature to decide upon the further course of action is an algorithmic approach that we considered. Krishnendu Rarhi et al [8] proposed using AIML Pattern detection to retrieve the symptoms of diseases that the user was suffering from, and using an Application Programming Interface (API) acting as an intermediary between client and server in order to deploy an appropriate response. Using the user input in sentences and utilizing a Chatbot engine that uses separate symptom and remedy arrays, which are then combined to give a solution to the end-user. Also, for testing purposes, the Authors suggested using General Word Percentage analysis, in which, the ratio of the number of unrelated words used over the total number of words was calculated in a message that was inputted by the user. This would enable verification of how the algorithm would deal in the event of the user giving unrelated words in addition to medical symptoms/terms, allowing for verification of overall algorithmic accuracy.

Sara Pérez-Soler et al [9] proposed automating the generation of modelling Chatbots for Domain-Specific Languages (DSLs), such that the implementation would take place over Google’s Dialogflow. Using Dialogflow would ensure optimal deployment and ensuring the streaming of data applications using a modelling Chatbot.

III. SURVEY

Methodologies have been established for conducting survey research aimed at ensuring that research is rigorous and robust outputs are obtained. While taking a survey, questionnaire validation and sample selection must be considered.

Websites and software are useful while drafting surveys. However, sources of bias may be inadvertently introduced. Positive attributes of online surveys include ease of speed and improved reach, along with ease of performing surveys and lower pricing slabs. Online surveys also allow for increased flexibility and automation. An online survey is performed such that participants quickly complete the questions asked in the survey. When the survey is disseminated through social media/email to individual survey takers, and completion incentives are offered, then respondents are likely to give a positive response [10]. Performing online surveys results in cost minimization and automation in capturing of responses, which reduces the need for paid researchers to question or enter information face-to-face, resulting in reduced input errors. As a result, data coding and cleaning have become increasingly obsolete.

Once the survey is completed, there are ways to download data into a selection of formats and imported it into analytical tool packages. We used Google Forms as a technology framework to perform a survey to determine the anxiety levels of people on a day-to-day basis, and whether or not people would prefer to have a virtual assistant to help them on daily basis. We had 75 respondents to our survey, all of whom belonged to diverse economic and social backgrounds. The questions in the survey questionnaire aimed at:

- Identifying potential end-users and their demographics.
- Ascertaining if survey respondents faced forms of depression/anxiety on a daily basis, and to what extent.
- Asking survey respondents if presence of Chatbot or conversational assistant will assist their mental health on day-to-day basis.
- Identifying price points which respondents would be prefer for facility of mental assistance Chatbot that tackles mental health on a day-to-day basis.
- Identifying potential end-user preferences on mobile-based or web-based applications.

Adam Palanica et al [11], implied that the main perceived drawbacks were the lack of capacity of Chatbots/conversational assistants to understand complicated emotions and handle requirements of end-user (patients). It was found that there was potential for significant risk, in the form of incorrect medical data, and wrongly programmed Chatbots responses Relevant data gathered is interpreted below in form of pie-charts. To examine characteristics of participant responses for Chatbot requirements in healthcare, the participant response data were analysed using descriptive statistics and frequencies.

With regards to facing workspace anxiety, 51.8% of the respondents reported facing minimal or no workspace anxiety at all, compared to 39.3% of the respondents who reported facing notable workspace anxiety, with the remaining 8.9% of respondents reporting some minor/contextual instances of facing workspace anxiety. This has been represented below in Fig. 1.

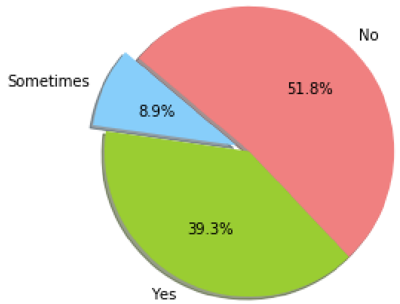


Fig. 1. Respondents reporting instances of workspace anxiety.

Likewise, when asked whether the presence of a virtual assistant will alleviate day-to-day anxiety, 25% of the respondents responded affirmatively, compared to the remaining 75% who felt that the presence of one would not make a significant impact on their day-to-day anxiety levels. This was expected, considering that most of the survey respondents had not used a Chatbot in their day-to-day lives before taking the survey. The pie-chart representation of this is shown below in Fig. 2.

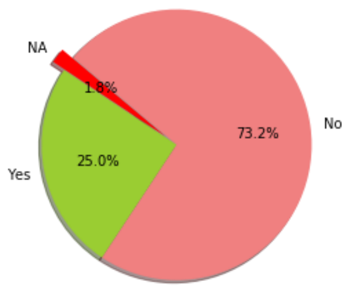


Fig. 2. Will a virtual assistant alleviate your day-to-day anxiety?

When asked about choosing between web or mobile-based applications, it was found that a notable majority of respondents (55.4%) preferred using a web-based application as opposed to 42.9% of the respondents who would prefer to download a mobile application. The remaining 1.8% of the respondents who responded stated that their preference between a web/mobile application will depend on the context. This has been represented in a pie-chart in Fig. 3.

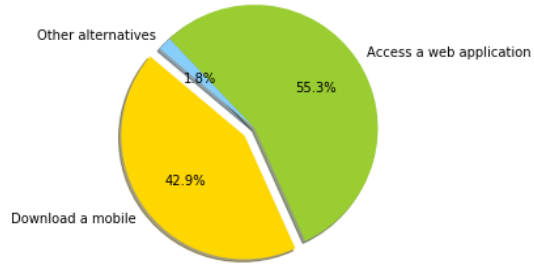


Fig. 3. Respondent preference between Web/Mobile Applications, amongst other alternatives.

Healthcare workers feel that they might utilize Chatbots to automate logical tasks but are unlikely to believe that Chatbots are sufficiently advanced to substitute complex decision-making roles that require expert medical opinion. Scanning through the survey results indicated to us that a mental-assistance Chatbot deployed as a web-based application would positively influence the lives of everyday users. Accordingly, we initiated building the conversational assistant and incorporating it into web-app using chosen technology stack.

IV. SYSTEM IMPLEMENTATION

In this section, we describe the process of system implementation. This section is structured into sub-sections. We start by elaborating on system architecture. Following that, we detail the implementation process, which is composed of 4 sub-stages. Eventually, we elaborate upon the toolset that was used to develop the conversational assistant.

A. System Architecture

Within DialogFlow, an intent classifies the intention of an end-user within a conversation instance [12]. For an individual agent, multiple intents with different roles can exist. The combination of these intents can perform a full conversation. Our conversational assistant functions as a component within the interface framework and employs an algorithmic approach that primarily utilizes intents. Dialogflow is used. to provide an interface for development of anticipated responses from a series of specific user queries.

We utilized Angular [13] as a front-end interface framework. Sanket Salvi et al [14] recommended Dialogflow for ease of implementation, coupled with a wide range of features and integration options. Nudtaporn Rosruen et al [15] proposed a mechanism that involved the utilization of Dialogflow as a chatbot response generator. The program sends messages via Dialogflow, and the intent is extracted from the message, and the response relayed to the end-user is pre-defined by the training phase of the intent.

Dialogflow is integrated into the Angular-based psySSIST, employing Content Delivery Network (CDN). The user-inputted message is subsequently extracted to obtain the message intent.The response given out by Dialogflow is as per the message intent obtained from user input previously. The intent produces actionable information conferring to dissimilar channels. When certainly matched intent is found by Dialogflow, a Webhook will deploy custom external APIs to discover a reply in external databases.

A Webhook is an agent that directs a configured answer to the intent [16]. Subsequent actions for the respective intent are described within the Webhook itself. The architecture of the algorithmic approach is shown below in Fig. 4.

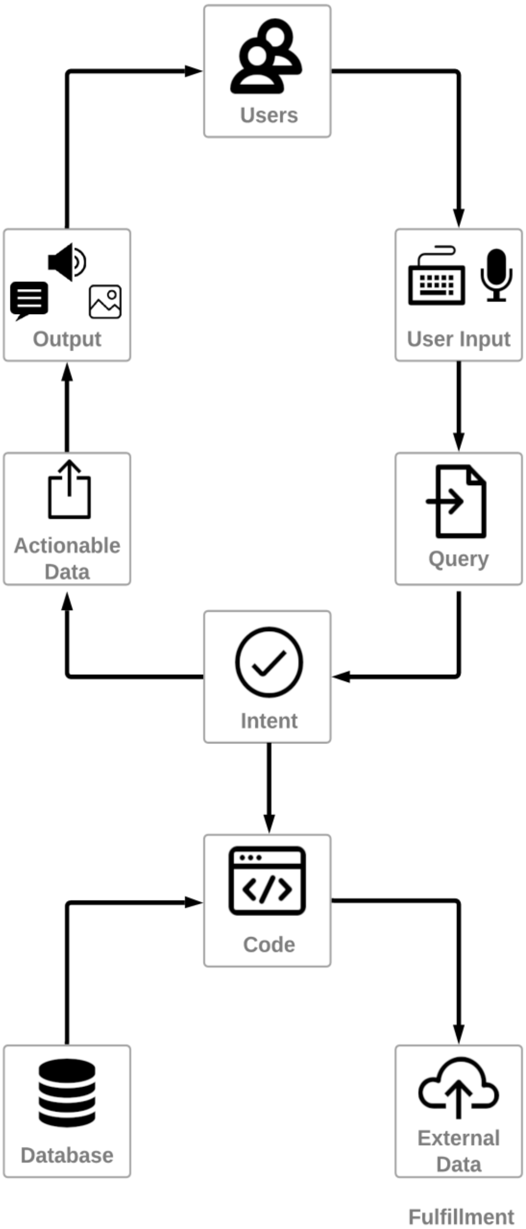


Fig. 4. Workflow diagram of psySSIST

For certain instances, to respond to the request message, the system will require additional data from external databases/APIs. Additional code base integration is required for this. Once these processes are performed, the psySSIST system will generate user-understandable actionable data. The application will relay this data to the user (in form of text/image/voice/video). The user should revert with a suitable response to continue the conversation.

B. Implementation Process

For implementation of psySSIST conversational assistant system, we consider following four stages which are:

- Stage 1: System Analysis.
- Stage 2: System Design and Use Case.
- Stage 3: System Development.
- Stage 4: System Assessment.

Detail of implementation process is shown below in Fig. 5.

Fig. 5. Implementation Process

1) System Analysis: Existing mental healthcare utility Chatbos were reviewed in order to examine them for relevant precedents which could be utilized to construct our own conversational assistant implementation. This was followed by looking up potential mental health disorders and their symptoms, along with pathways required to move forward once a reliable diagnostic assessment has been obtained. Symptoms that psySSIST was trained to look out on a daily periodic basis includes anxiety/depression along with their clinical severity levels. In later stages, the previously obtained information is used as a Knowledge Base (KB) in order to train the conversational assistant. From the survey and subsequent analysis, we took into account a few limitations that we had stumbled upon. These limitations could impact a user's overall usage experience of psySSIST system. The limitations were as follows:

- User might not know how to install/access the application.
- User might not know how to use the application.
- Application requires compatibility with different device configurations (irrespective of device platform/ screen size/processing power).

To deliver cross-platform computing experience across wide range of devices, psySSIST was developed as web application using Angular framework, resulting in minimized application learning curve for common for all users.

2) System Design & Use Case: In this section, we shall detail the system design of psySSIST, and the use case.

a) System Design: psySSIST conversational assistant system utilizes a total of 28 intents, which were trained by means of using Dialogflow. These are composed of 12 intents (each of which describes user symptoms), followed by 10 intents (each of which details questions which monitor for day-to-day anxiety). The following 3 intents (each comprised of sub-details of headaches/seizures/migranes) are present. Lastly, 3 intents dealing with (user greetings/no illness symptoms present in user/finding local healthcare workers). psySSIST can serve as a substitute for those that require interaction with mental health professionals on a day-to-day basis, such that it can suggest and provide mental health inputs to those individuals that require them.

b) Use Case: Intended use case for psySSIST as a conversational assistant, is to provide generalized consulting for psychiatric conditions. By virtue of clinical severity, psySSIST algorithm would interact with user and determine if said symptoms correspond to what is present in the symptom database. If the symptoms require clinical intervention, the user would be linked to the closest healthcare worker, considering the user preferences (which include location/age/gender/linguistics).

3) System Development: An Angular project was created order to develop psySSIST as a conversational assistant. Upon creating an Angular project, an initialize routine allows us to link different components of the application. Individual components are equated with a specific access URL. Plug-ins (such as Bootstrap/jQuery/Firebase) are imported and deployed within the project by means of Content Delivery Network (CDN). Doing so reduces delays in loading web page content by allowing quicker component rendering using fewer lines of code, resulting in better code optimization. To deploy the intent-based algorithm, Dialogflow was used as platform to simulate back-end functionalities. The stages of this process are as follows:

- Create a new agent (psySSIST) in Dialogflow on Google Cloud Platform using default Google Account credentials.
- Specify 28 intents that monitor user information (monitoring attributes like symptoms/day-to-day anxiety/headaches/seizures/migraines/greetings/sick ness/asking user location preferences) in psySSIST agent.
- Individual phrases and responses corresponding to them are to be paired together as functions in intents, and be mapped with user inputs.
- Describe and map responses (in phrases, image, voice, and video format) that are to be displayed to the user.

Utilizing JSON format in back-end functions to store user information as a JavaScript object is necessary, to relay responses back to the user to-and-from the Dialogflow server. User symptoms are inputted into phrases during training process in order to improve learning rate accuracy of the intent-based algorithm. An instance of performing phrase training is shown below in Fig. 6.

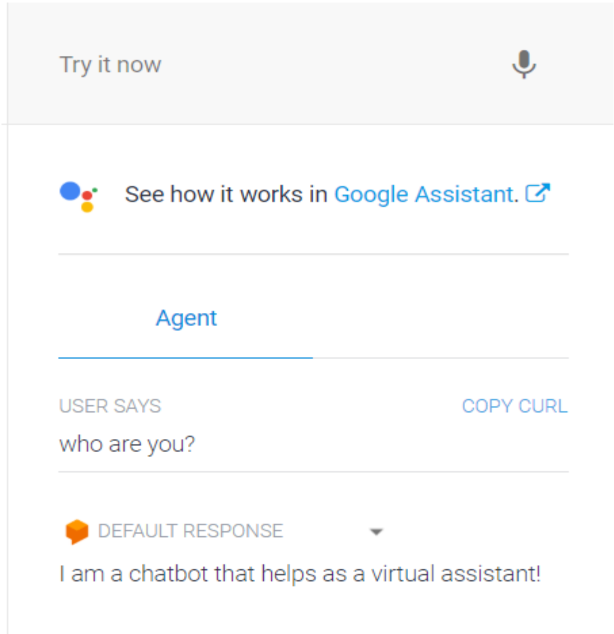


Fig. 6. Training agents in Dialogflow

4) System Assessment: Performing testing, assessment and evaluation of the components present in psySSIST conversational assistant project is final step of implementation. This is performed in order to ensure system performance matches the required parameters. After this, it is ready for deployment.

There are 2 phases to this step, which are as follows:

- System assessment during training.
- System assessment after training.

Assessing the system during training is performed to verify phrases of conversation in a given situation and check if their corresponding responses are appropriate considering the situation. More training phrases are to be appended in the event that the response to the said situation is incorrect. Utilizing in-built consoles present in Dialogflow (which incorporate conversation context ranging from dialogue history, location and user preferences), we can verify functioning of individual intents. Once system development is completed, another round of system assessment should be performed. Conversation is simulated by setup and test. For testing, the Chatbot platform uses conversation phrases that are not used to train the Chatbot.

C. Development Toolset

Dialogflow (formerly known as api.ai) is used as the engine for the conversational agent in this prototype. It is a product for Chatbot development which benefits from support of Natural Language Understanding (NLU) to help create a chatbot without algorithmically coding each and every sub-component. Using NLU, Dialogflow can convert input/query into intent. If user-inputted query is not precisely matched, in-built Machine Learning (ML) resources are utilized by Dialogflow. These resources are combined with native Natural Language Processing (NLP) algorithms in order to interpret the user-inputted query. Compared to conventional conversational agents, users here have ability to use limited/smaller inputs to generate input query. Entities are identified from user-inputted query, and are classified on basis of their working. These identified entities are subsequently assigned with a corresponding action, which is triggered when relevant query is inputted by the user. Conversational agent entities are then submitted to Webhook for request fulfillment.

For constructing conversational agents, Dialogflow has a set of features that create various Chatbot modules. The specifics of the features are defined below:

1) Agent: In order to comprehend user input, this module is used in combination with Natural Language Processing (NLP) algorithm integrations. Initialization for this module should occur when conversational agent is used.

2) Intent: It is module that classifies intention of an end-user within a conversation instance It is utilized in order to support/determine further course of action. The intent module includes the following sub-modules:

a) Context: It is attribute that is utilized in order to remember passing intent.

b) Event: It is a substitute process by which intent can be triggered. It can also define new events.

c) Training Phrase: On considering training phrase "My car is red in color", phrase trains the agent module to recognize end-user expressions that are similar to that phrase, such as "what color is my car?" or "red car". Training Phrases must be composed of atleast 15 examples in order to improve accuracy.

d) Action parameters: These are parameters that are depict specific information such as name/location/ date/ time.

e) Response: Message that is relayed back to the user by conversational agent after user gives input.

3) Fulfillment: It is function that represents a complete instance of conversation between conversational agent and user, such that request and response pairs are passed on between both sides.

4) Integration: These are tools by which conversational agents are integrated with other popular platforms for instance Assistant/Slack/Messenger/Twitter/WhatsApp.

V. TESTING

Seen in this section are the observations noted during testing/deployment along with overall system test results. The efficiency rates and response times of psySSIST were noted in real-world scenarios to observe the response time. The reasoning behind performing system testing during training process is to verify the responses of conversational agent to a given real-world conversational situation. In the event of incorrect responses, appending new training phrases (relevant to the situation) might be required to ensure dynamic responses from Dialogflow.

The authentication protocol that allows end-user to log in from a single e-mail ID and password page is called Single Sign-On (SSO). This protocol was implemented in psySSIST by using Lightweight Directory Access Protocol (LDAP) to store databases in the server directory. Users were offered the opportunity to enable analytics, such that a JavaScript function enables the analytics function present in the angular.json file present within the Angular project. The analytics function was implemented such that user data is stored anonymously in order to avoid data abuse.

Demonstrated below, in Fig. 7, the payload input was taken from the user end, and the Chromium Browser reads the data payload submitted by the user. This is subsequently sent to a server.

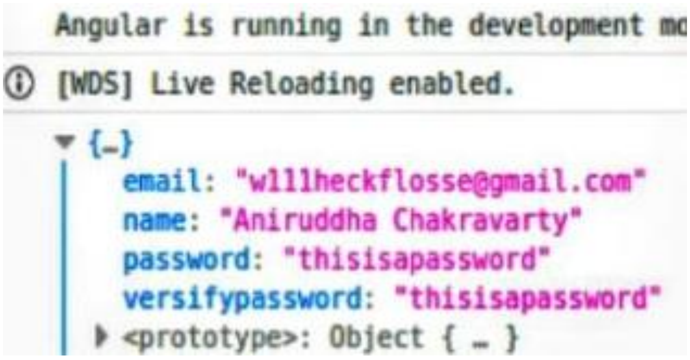


Fig. 7. Payload of user registration details, seen from a Mozilla Firefox 76.0 Development Console.

The chat component does not reload the page each time a new message is inputted by the user, or a new response comes from the Dialogflow API via the CDN. This is done by adding JavaScript XMLHttpRequest() function to chat component, such that new messages are displayed on the page as user scrolls down the page. The chat component was deployed in a variety of environments including iPhone (SE/6/7) and Android based Google Pixel (3A/XL) and Linux PC.

Shown below, in Fig. 8 is the implementation of the WebKit-scrollbar, which is a plugin supported by the Mozilla Developer Network. We defined the scrollbar button, thumb, track, track piece, along with a resizer that allows the scrollbar to size itself appropriately in different browser environments and configurations across different devices.

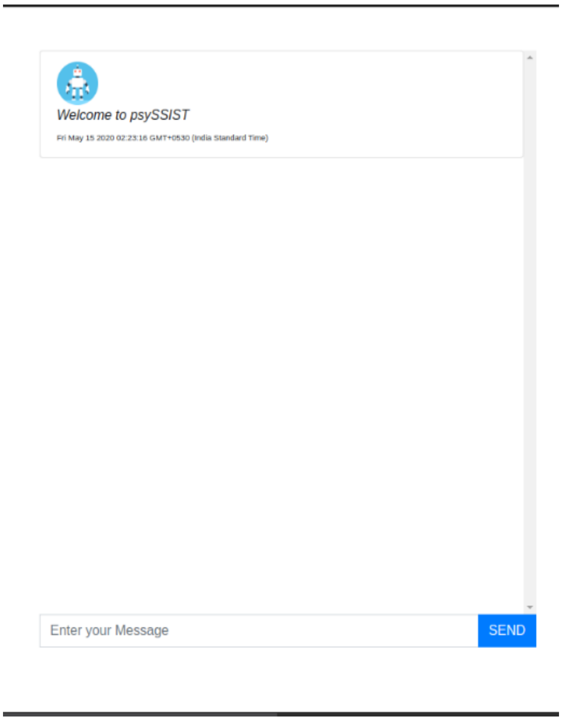
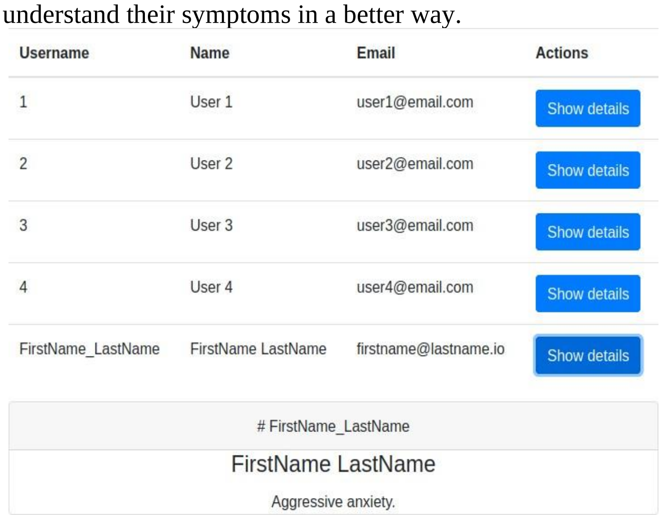


Fig. 8. Integration of a scroll bar on the psySSIST chat component using WebKit-scrollbar.

Seen in Fig. 9 we find that user database is created such that doctors can track the patient's previous records to help understand their symptoms in a better way.



Username	Name	Email	Actions
1	User 1	user1@email.com	Show details
2	User 2	user2@email.com	Show details
3	User 3	user3@email.com	Show details
4	User 4	user4@email.com	Show details
FirstName_LastName	FirstName LastName	firstname@lastname.io	Show details

FirstName_LastName
FirstName LastName
Aggressive anxiety.

Fig. 9. Database of all users

Each user case has previous medical history voluntarily saved of the user to fill up details on his/her medical history. If the user's history is present, the practitioners will be able to solve cases more rapidly and identify superior treatments.

Characteristics of data required to be inputted by end-user includes ease of data collection from mobile devices. Changes made to the input values constitute significant activity variables, which may cause change in status of mental health diagnosis for end user. Joo-Chang Kim et al propose a user behavior prediction model [17] that utilizes Recurrent Neural Networks (RNNs) in combination with long short-term memory of user-inputted data characteristics (DC-LSTM). In terms of selecting variables for the model, attributes and surrounding context of data obtained from mobile devices used by end-users were scrutinized in order to provide customized services to the same end-user. Integrations within Dialogflow that are capable of collecting, supplying, processing and analyzing large-scale data received from mobile devices in RSS and XML formats are being developed as Knowledge Bases (KB).

The User Interface of the psySSIST chat component is shown in Fig. 10, below.

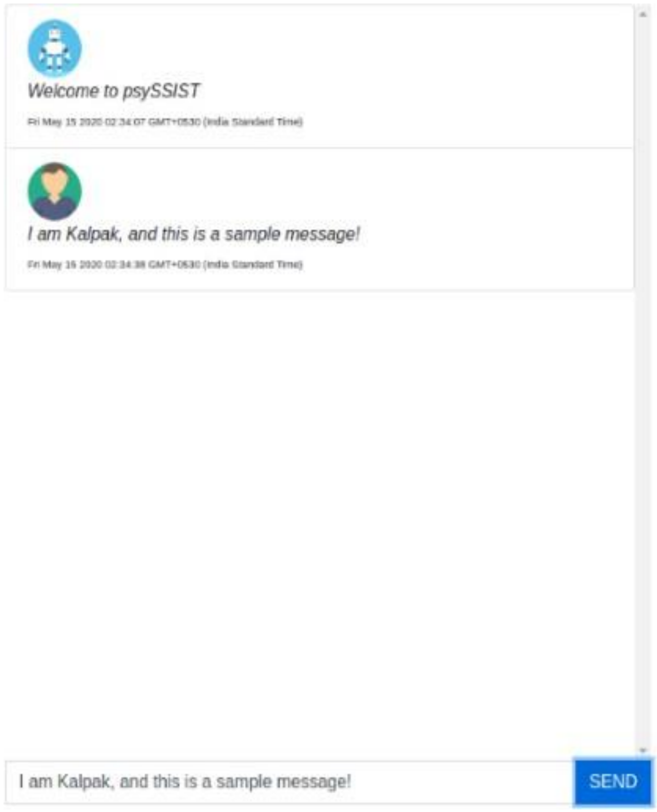


Fig. 10. User Interface of psySSIST chat component.

David Vronay et al [18] suggest thinking of chat in terms of a real-time data type that is capable of streaming media (in text/audio/video formats) between 2 users that are on either side of a single channel. This real-time data type would contain indicators that denote the status of the channel, such that the indicators would contain diagnostic information about the status of the chat and channel, along with user activity logs. As a result, troubleshooting problems commonly faced by computer-based text chat users would become significantly easier.

VI. CONCLUSION & FUTURE SCOPE

A. *Conclusion*

By means of this paper, we proposed an approach to define a Natural Language (NL) based conversational syntax, which was then applied to develop an automated mental health conversational assistant that lays out personalized symptom recognition based on user input. This approach is based on performing annotations on domain meta-models, with configuration information for Natural Language (NL) Syntax and translating this into a Chatbot creating framework.

The prototype demonstrates the modelling of a conversational agent atop an existing cloud system to define and run streaming data applications. The functionalities added by the Chatbot, are illustrated, which include support for collaboration in Natural Language (NL) along with multi-platform mobility and traceability.

We are able to successfully mitigate the symptoms of mental disorders and allow the end-user a certain sense of functional motivation which allows him to function as an effective individual.

B. Future Scope

This paper describes programming architecture of conversational agent web-apps in context of end-user mental health. Assessment of the quality and usability of our generated Chatbots is still ongoing. Subsequently, in future, we intend to perform a usability study with users, and develop quality frameworks for conversational assistants that are suited for healthcare domain.

ACKNOWLEDGEMENT

We dedicate this paper to the 264 million individuals who suffer silently from undiagnosed chronic depression.

In addition, we remain grateful to all essential healthcare workers/doctors and unsung heroes who risked their lives for society during the 2019-20 COVID (SARS-CoV-2) pandemic.

REFERENCES

[1] Nagarhalli, Tatwadarshi P., Vinod Vaze, and N. K. Rana. "A Review of Current Trends in the Development of Chatbot Systems." 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE, 2020.

[2] Vaidyam, Aditya Nrusimha, et al. "Chatbots and conversational agents in mental health: a review of the psychiatric landscape." The Canadian Journal of Psychiatry 64.7 (2019): 456-464.

[3] Google. Dialogflow. https://dialogflow.com/, 2021.

[4] Ranoliya, Bhavika R., Nidhi Raghuwanshi, and Sanjay Singh. "Chatbot for university related FAQs." 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE, 2017.

[5] Thomas, N. T. "An e-business chatbot using AIML and LSA." 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE, 2016.

[6] Oh, Kyo-Joong, et al. "A chatbot for psychiatric counseling in mental healthcare service based on emotional dialogue analysis and sentence generation." 2017 18th IEEE International Conference on Mobile Data Management (MDM). IEEE, 2017.

[7] Hien, Ho Thao, et al. "Intelligent assistants in higher-education environments: the FIT-EBot, a chatbot for administrative and learning support." Proceedings of the Ninth International Symposium on Information and Communication Technology. 2018.

[8] Rarhi, Krishnendu, et al. "Automated Medical Chatbot." Available at SSRN 3090881 (2017).

[9] Pérez-Soler, Sara, et al. "Towards Conversational Syntax for Domain-Specific Languages using Chatbots." J. Object Technol. 18.2 (2019): 5-1.

[10] Ball, Helen L. "Conducting online surveys." Journal of Human Lactation 35.3 (2019): 413-417.

[11] Palanica, Adam, et al. "Physicians' perceptions of chatbots in health care: cross-sectional web-based survey." Journal of medical Internet research 21.4 (2019): e12887.0

[12] Google Cloud. Dialogflow ES Documentation. https://cloud.google.com/dialogflow/es/docs/intents-overview, 2021

[13] Angular. https://angular.io/, 2021

[14] Salvi, Sanket, V. Geetha, and S. Sowmya Kamath. "Jamura: A Conversational Smart Home Assistant Built on Telegram and Google Dialogflow." TENCON 2019-2019 IEEE Region 10 Conference (TENCON). IEEE, 2019.

[15] Rosruen, Nudtaporn, and Taweesak Samanchuen. "Chatbot utilization for medical consultant system." 2018 3rd technology innovation management and engineering science international conference (TIMES-iCON). IEEE, 2018.

[16] Sujana, A. S., DR Ramesh Babu, and S. Venkatesan. "Twilio Integration with Dialogflow for Effective Communication." Journal of Web Development and Web Designing 4.2.

[17] Kim, Joo-Chang, and Kyungyong Chung. "Prediction Model of User Physical Activity using Data Characteristics-based Long Short-term Memory Recurrent Neural Networks." KSII Transactions on Internet & Information Systems 13.4 (2019).

[18] Vronay, David, Marc Smith, and Steven Drucker. "Alternative interfaces for chat." Proceedings of the 12th annual ACM symposium on User interface software and technology. 1999.