GuidoBot : A Novel Approach for Assisting Faculty in Tracking Goals and finding tasks

¹ Nagothu. Srujan, ² Utlapalli. Mahesh, ³ Vemuri. Poojitha, ⁴ Syam. Jason,
Jonnalagadda. Surya Kiran

¹,²,³,⁴ Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,
Vaddeswaram, A.P., India-522502.

¹ srujannagothu99@gmail.com , ² umahesh987654321@gmail.com , ³ poojitha.vemuri20@gmail.com ,
⁴ syamjasonb777@gmail.com , ⁵ kiransurya93@kluniversity.in

Abstract: In the world of Artificial intelligence we have many inventions that facilitate human works, one of such innovation is a Chabot. We are developing a Window based Guido Bot as a guide which helps faculty for Goal tracking and in managing daily tasks effectively. We are designing this Chabot with three different components like Profile Maintenance module which deals with the academic profile, status of research publications of faculty. And an important module is a Goal Tracking module that helps faculty to achieve their academic goals, research goals with timely updates and notifications. And the Task Finding module helps faculty in assigning and tracking the tasks status with daily updates of pending tasks along with their deadlines. And we are developing a web application as an interface for admins (Management) to monitor and issue goals and administrative tasks for faculty. And the Guido Bot uses this interface as a data source for efficient performance.

Keywords: Bot, Goal Tracking, Profile Maintenance, Administrative tasks, Academic goals, Task Finding.

I. INTRODUCTION

A chatbot is one of the most important game-changing innovations in the field of AI and ML during the last decade. A Chatbot is a Virtual Enterprise software application with a Conversational User Interface (CUI) that provides diverse services in the fields of health, education, management, and Government services. It permits various types of interchanges between a human and a machine, by means of messages or voice order. A chatbot is modified to work freely from a human administrator. It can answer questions planned in regular language and react like a genuine individual. Chatbot revolution, hence brought communication, to an entirely new customized level. Today's chatbots, further developed since then, are ready to answer considerably complex questions [13] in an effective timely manner and able to automate human tasks efficiently. There are two types of Chatbots: i) Simple Chatbots – Chatbots that function on predefined rules and can answer to only limited queries. To find an answering user needs to make several selections [14]. If a user asks a question that doesn't match with the database present, the chatbot would be unable to answer the query and print a message "sorry I don't know" as these chatbot's inbuilt code could solve only simple use cases. ii) Smart Chatbots – Chatbots that rely on Machine Learning and Normal Language to know the user's language which is intelligent enough to remember conversations with specific users to learn and grow over time in a dynamic environment. Instead of static answers, these chatbots respond with several other recommendations based on the previous chat data and data from similar user personalities [15]. These chatbots are built on code that is complex in nature. Smart Chatbots have endless applications. With the help of technology, we had seen considerable innovation to track and manage individual and group tasks, i.e. through reminder apps, email, messenger, etc. In this current AI world, we have a wide variety of chatbot applications that infiltrate every corner in educational institutions that assisting Management and students for effective interactions and querying well about the institution in real-time. There is no existing system that provides assistance for faculty in order to achieve their goals in a timely manner. In a few Applications User's need to write complex queries in order to get the desired result or users need to refer to the web portal (ERP) every time which is time-consuming. Also, there is an ambiguity regarding timings scheduled for classes and events. So, to overcome this problem there is a need for a system that guides the faculty in order to create, assign, manage, track, and accomplish their goals and objectives in the desired time frame. chatbots are profoundly accessible, incredibly refine, and can be a partner for faculty. Groups that take on chatbots will end up approaching up to effectively achieving the numerous impossible assignments and discrepant objectives that their work requests.

II. LITERATURE SURVEY

A. Title: A Question Answering and Quiz Generation Chatbot for Education

Authors: A.S. Sreelakshmi, S.B. Abhinaya, Aishwarya Nair, S. Jaya Nirmala

In this paper, Chatbot performs three tasks i.e. answering user questions, generating a quiz for the user, evaluating the performance of the user. The user provides documents as input to the Chatbot. This input undergoes several pre-processing [16] steps and acts as an expert system and this expert system is taken as input for the first module as well as

the second module. The first module consists of ranking functions to take out the top 10 queries from the expert system by using the BM25 score. These queries are given as input to the neural network. This output of the neural network will be the answer given by the chatbot to the user questions. The second module ranks sentences based on their significance to the content in the expert system. Relevant sentences from the expert system are modified using NLP and are framed as questions to the user quiz. It is assessed based on the Answerability of a question and the Suitability of a sentence. Quiz answers given by the user are given as input to the third module i.e. Answer Comparison module. Its outputs score indicating the performance of the user Main drawbacks are feedback-based improvement is not present in order to generate the user level of understanding of the question. Every time user needs to upload the required documents/pdf's/pics to the chatbot. Complex Questions can be answered by consolidating the required data from all the uploaded pdfs.[1]

B. Title: Intelligent Chatbot for Guided Navigation of Repository Contend

Authors: Anjali Mishra, Shruti Sapre, Shruti Shinde, S.N Shelke

This paper presents an approach for Answering user questions for particular content. In the beginning, the user gets registered, and then after the successful registration process, the user uploads the file in NLP format. The uploaded file gets trained by the system and get saved in the database used in this application. Then the user can ask a question to the chatbot system that he/she want the answer and the user question is considered as an input. All the input related questions within the document are considered as streaming questions [17]. User input and streaming questions are taken into a queue. The system then selects the object from the user question and matches it with the streaming question by reading the file content and splitting the user input question line by line to match with the streaming questions and generate an answer. The generated answer gets appended with the question and gets displayed to the user. The proposed system can be used by farmers and also at soil investigation centers. The main drawback is the document type is only one specific type and this application can be used by only one user at a time. [2]

C. Title: Adoption of AI-Chatbots to Enhance Student Learning Experience in Higher Education in India

Authors: Nitirajsingh Sandu, Ergun Gide

This paper discusses how Chatbot technology can improvise student learning
 as in the same classroom, each student has different learning needs and interests. Therefore, each one can use the help of a specialized guide. It is common for students at all levels to use some kind of messaging service to communicate with each other and, occasionally, with their teachers but they are not available all the time. Students recognized that Chatbots respond in a faster mode as they are available 24/7. All the

educational issues faced by the students would get clarified easily in a short span of time by using chatbots. They gave the highest response time when compared [18]. It also helped students in paying their fees. Immense benefits can be achieved by the students with the usage of a chatbot for higher education i.e. Student-centered learning. On the other hand, users also had a doubt of losing personal information when submitted to the chatbot when required. Users received inappropriate advice from the chatbot as sometimes the chatbot could not process the user input properly. User addiction to the chatbot might create a negative effect. [3]

D. Title: Chatbot and bullyfree chat

Authors: V. Selvi, S. Saranya, K. Chidida, R. Abarna

It is impossible for humans to go through all the texts from conversations available online manually to determine if there is a problem. So, this paper discusses how NLP can be used in an efficient way to understand the user text and perform sentiment analysis to detect bully words. Sentiment analysis is generally denoted as techniques used to determine the predisposition of text, usually expressed in free text form. A sentence is parsed using NLP. Bully words are present in three forms i.e. in texts, user demography, and social networks features. Bully space is constructed. SDA concept is used to detect cyberbullying. A supervised machine learning algorithm is used to check bot performance and continuously trained to enhance its performance. The advantage is that chatbot can get required information according to the user queries by automatic removal of bully words. The main drawback is the usage of the AdaBoost M1 algorithm as its performance is low when the noise in the dataset is high. [4]

E. Title: Possibilities of used intelligence-based agents in instant messaging on e-government services

Authors: Dusko Sivcevic, Ivan Kosanin, Slobodan Nedeljkovic, Vojkan Nikolic

This paper discusses how web agents can be used in both the individual, business, and all government sectors to offer better services. In this framework the deployment of artificial intelligence systems can provide transparency in communication to optimize government functionalities like the execution of ongoing policies, passing files to various sectors, delivering services, maintaining law & order. Automated Systems gives the highest response time to all the users. Applications are linked to instant messaging services like Viber allowing citizens to get address their issues. The benefits of web agents in governance are It simplifies the process of assembling and collecting government information regarding any department to the citizens and business. Ease of availability of police services 24*7 for every citizen through online Agents. [5]

F. Title: Artificial intelligence in warehouse automation for flexible material handling

Authors: *Hendrik Thamer, Axel Börold, Ariandy Yoga Benggolo, Michael Freitag*

This paper discusses how to improve transferability among the objects. Various sets of image samples of a particular

class are considered, and these samples are trained to automatically detect how object class can contain its feature description. Based on this system can classify past unrecognized images and present familiar objects. With the help of deep learning [10] the system will be able to function in the right manner even though the user gives invalid inputs or in a dynamic environment. We can also overcome the problems regarding training data as the system now can identify all the parts of the image and multiple movements and positions of user images are considered. This approach can also be used for the objects presented in multiple platforms. Noise detection in the data can be improvised. [6]

G. Title: A Survey on Chat-Bot system for Agriculture Domain

Authors: Prashant Y. Niranjan, Vijay S. Rajpurohit, Rasika Malgi

This paper discusses how chatbots can be implemented in the agriculture domain. Farmers can ask their questions related to weather changes, raw materials, plant protection methods [12]**, best** practices adopted, etc. Farmers can utilize chatbots through mobile as it acts as an effective tool to provide efficient information access in a rural environment for people with less literacy rate and technology knowledge. Chatbot consists of 3 Modules. The first module is Question analysis second module is document processing [20] and answer extraction being the third module. Initially, the user query is processed using nlp techniques. In the second step i.e. document processing, all the query-related documents in the web are searched by using RNN (Recurrent neural network) as it consumes less time. The data set is in XML format. It consists of all the previous questions asked by the farmer indicating continuous learning by the chatbot.[7]

H. Title: A Chatbot for Psychiatric Counseling in Mental Healthcare Service Based on Emotional Dialogue Analysis and Sentence Generation

Authors: Kyo-Joong Oh, Dongkun Lee, Byungsoo Ko, Ho-Jin Choi

Today, the need for mental health [11] services is increased. This paper introduces a chatbot that can be used for mental healthcare. It recommends that continuous emotional recognition and the ethical response of the user need to be observed. It helps in notifying the user status when emergency situations occur. Chatbot gives the quickest response to the user's family members when required and can help in saving a life. Continuous Observations for a critical disease are not noted by the chatbot but helps the users in reducing their mood swings in the form of dialogues. people in distress can reach out. The need for one-on-one, individualized services have replaced other platforms. The user who needs psychiatric counseling can input contents to the chatbot in the form of text or the user can send images, videos, voice messages, etc. It checks the emotional imbalance as well as user problems and stores it in its log records of the user and gets updated continuously.[8]

I. Title: Chat bot Using API: Human To Machine Conversation

Authors: Sunil Punjabi, Vighnesh Scthuram, Vignesh Ramachandran, Ronit boddu, Shivshankar Ravi

The main motive of this paper is to resolve the issues faced by the people using the IRCTC website. Input by the user is taken in the form of text or voice and is processed using NLP and accurate information is provided to the user regarding IRCTC. All the people using the site for ticket booking, checking seat availability, PNR status can get their queries solved quickly. All the services of the IRCTC website are provided under one roof. The chatbot is efficiently worked by combining with AI lead platforms like Google assistant and social media platforms like Facebook, etc. It is user-friendly as it can be used through any social networking site without any external installations. Response time of chatbot is high compared to MakeMyTrip bot. There are 4 phases in this Chatbot. Initially, dialog flow is used for creating entities and actions for the system. In the next phase with the help of java server-side coding is implemented. Further postman is used for testing and making a post request. Lastly, railway site API is used to provide accurate output.[9]

III. METHODOLOGY

Guido framework consists of 3 modules
- i. Profile Management
- ii. Goal Tracking
- iii. Task Finding

Module 1: In **Profile Management** faculty can view their academic and research Information along with their awards and recognitions, and there is also a provision for them to update their profile using this Guido Bot.

Module 2: **Goal tracking** module helps to accomplish research, and Academic goals of faculty. Based upon role, goal tracking module has different functionalities i.e. based upon role of Assigner and Assignee.

For Assignee, he/she can view his/her pending goals and their deadlines and can upload the work that he/she completed as a progress.

For Assigner, he/she can verify the work of assignee and can update their progress of the goals based on his/her completion of goal.

Module 3: **Task Finding** module helps to accomplish administrative tasks, academic and research tasks of faculty.

For Assignee, he/she can view his/her pending tasks and their deadlines and can update their task status (Inprogress / Completed) and can view the acceptance of their task from the assigner.

For Assigner, he/she can assign the tasks and able to track their status also have a privilege to accept or reject the task done by assignee.

```
Algorithm 1 GoalTracking
    INPUT:Set of Goals Assigned (G),Id of Goal (ID ) to be updated
    OUTPUT:Goal status (S)
    Initialize: G = Goals Assigned
              P = Progress of Goal (0 ≤P≤100)
              S = Status of goal
              A = Acceptence of Assignee
              ID = Id of goal to be Update
 1: procedure GOAL(G,ID):
 2:     for each G do
 3:         if (Id of G ≡ ID) then
 4:             A ← Acceptence
 5:             if (A ≡ "Accepted") then
 6:                 Uploading files of completed Work
 7:                 P ← Progress by Assigner
 8:                 if (P = 100) then
 9:                     S ← "Completed Goal"
10:                 else if (P ≤99) then
11:                     S ← "Under Progress"
12:                 end if
13:             else if (A ≡ "Rejected") then
14:                 S ← "Rejected Goal"
15:                 break;
16:             end if
17:         else
18:             Continue;
19:         end if
20:     end for
21:     return S
```

Fig 1: Algorithm for Goal Tracking.

Algorithm 1: Goal Tracking

Step 1: Among all the goals (G) assigned to him, assignee needs to select a particular goal using its goal id (Gid).
Step 2: If he accepts to work on that goal then he needs to change its status to accepted.
Step 3: He need to upload all the work that he progressed as a file. Assigner will verify it and then update the progress.
Step 4: If the progress reaches to 100% then the goal will be considered as completed else it will be considered as under progress.
Step 5: If he rejects to work on that goal then he needs to change the status to rejected.


```
Algorithm 2 TaskFinding
INPUT:Set of Tasks Assigned (T),Id of Task (ID ) to be updated

OUTPUT:Task status (S)

Initialize: T = Assigned Tasks
            S = Status of Task
            A =Approval by Assigner
            U= Updated task state by Assignee
            ID = Id of Task to be updated
 1: procedure TASK(T,ID):
 2:     for each T do
 3:         if (Id of T ==ID) then
 4:             U ← Update Task state
 5:             if (U ≡ "In Progress") then
 6:                 S ← "InProgress"
 7:             else if (U ≡ "Completed")  then
 8:                 S ← "Completed"
 9:                 if (S ≡ Completed) then
10:                     A ← Assigner Approval
11:                     if (A ≡ "Approved")  then
12:                         S ← "Task Approved"
13:                     else if A ≡ "Rejected" then
14:                         S ← "Task Rejected"
15:                     end if
16:                 end if
17:             end if
18:         else
19:             continue;
20:         end if
21:     end for
22:     return S
```

Fig 2: Algorithm for Task Finding.

Algorithm 2: Task Finding

Step 1: Among all the tasks (T) assigned to him, assignee needs to select a particular task using its task id (Tid).
Step 2: Initially the status of the task is in assigned state and when assignee started working, he needs to change the status as Inprogress.
Step 3: If he completes the task, he needs to update the status as completed.
Step 4: Once the status is in completed state the assigner will verify the task done by the assignee and either he can approve or reject the task submitted.
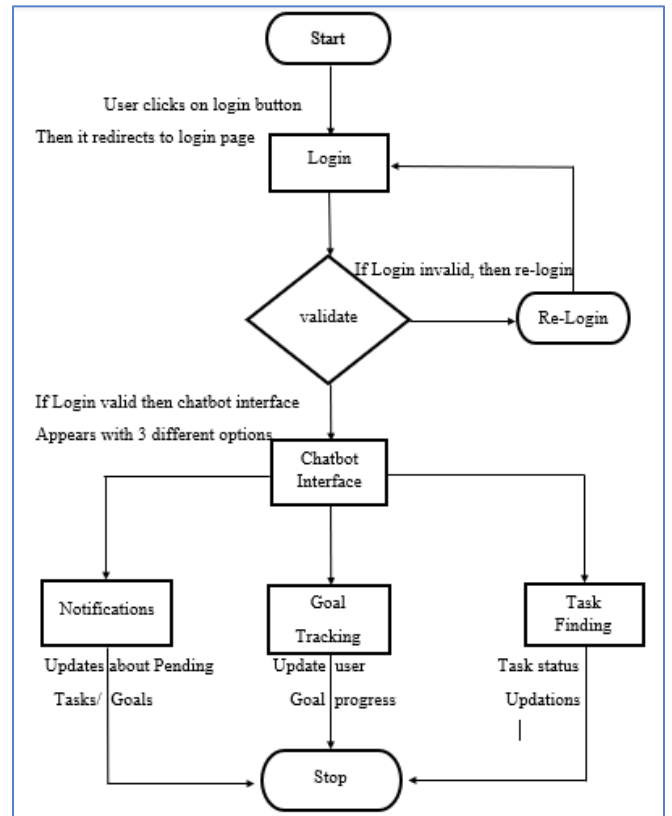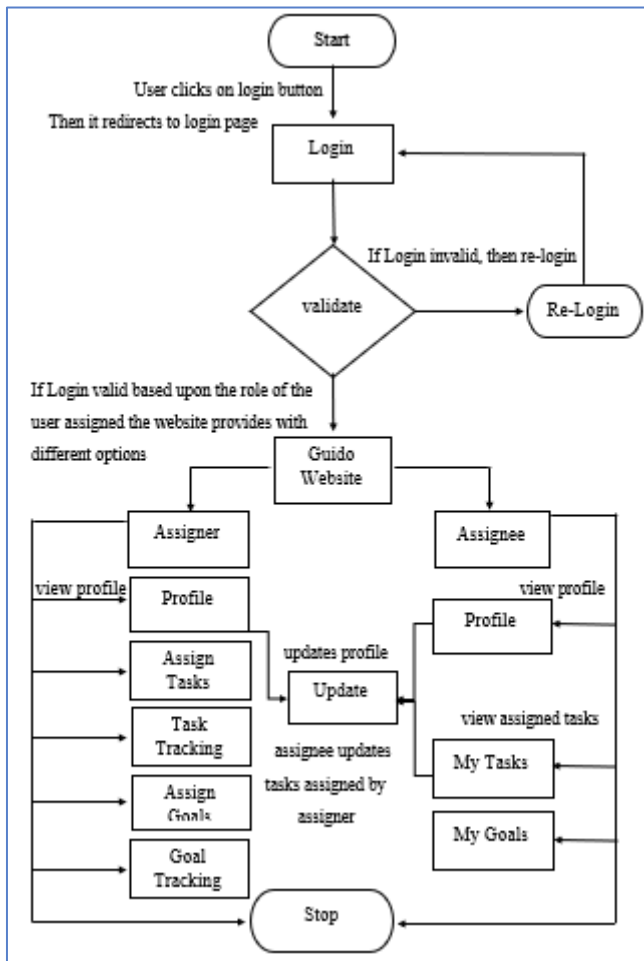
IV. BLOCK DIAGRAM

Fig 3: Block Diagram of Chatbot

Workflow of the Guido Chatbot:

Step 1: If a Faculty is already a registered user, then he/she can sign in to the chatbot, or else the faculty need to get registered with the help of admin.

Step 2: When the Faculty sign in with the correct login credentials then they will be redirected to the Interface of the chatbot or else they will be redirected to the login page again.

Step 3: In the interface users have an option to open the Notifications, Chat, or Task Finding tab based on his/her interest.

Step 4: If the user opens Notifications tab then he/she is having different types of notifications that is pending goals

and assigned tasks. Notifications can help user from not missing any of his tasks in his hectic schedule.

Step 5: If the user opens Chat tab then he/she can perform task updates, can change his/her goal progress that will be reflected with the admin, and manage their profile updates.

Step 6: If the user opens Task Finding tab then he/she can view his/her pending tasks and their deadlines and even he can update his task status using GuidoBot.

Step 7: Based on the user interest he/she can interact with the chatbot and can resolve all his queries instantly.

Fig 4: Block Diagram of Guido Website

Workflow of the Guido Website:
Step 1: If a Faculty is already a registered user, then he/she can sign in to the chatbot, or else the faculty need to get registered with the help of admin.

Step 2: When the Faculty sign in with the correct login credentials then they will be redirected to the Home page (Landing Page) of the website or else they will be redirected to the login page again.

Step 3: Now based upon his/her role of the user assigned the Guido website provides different views, which is the Assignee view or Assigner view.

Step 4: If the Assignee is the user then the assignee is having different options in his/her dashboard.

  1) Profile
  2) Assign Tasks
  3) Task Tracking
  4) Assign Goals
  5) Goal Tracking

    a) In Profile he/she can view the profile and manage their profile updates.

    b) In Assign Tasks he/she can assign tasks to the assigner.

    c) In Task Tracking he/she can track the tasks that are already assigned by the assignee, and even change his/her goal progress.

    d) In Assign Goals he/she can assign goals to the assigner.

    e) In Goal Tracking he/she can track the goals that are already assigned by the assignee, and even change his/her goal progress.

Step 5: If the Assigner is the user then the assigner is having different options in his/her dashboard.
  1. Profile
  2. My Tasks
  3. My Goals

    f) In Profile he/she can view the profile and manage their profile updates.

    g) In My Tasks he/she can check the tasks that are assigned by the assignee, and even change his/her goal progress.

    h) In My Goals he/she can check the goals that are assigned by the assignee, and even change his/her goal progress.

V. IMPLEMENTATION

A. Python: Tools, Libraries, Usage
- Spyder is the Scientific Development Environment for Python (v3.8), it is a free integrated development environment (IDE) which is integrated with Anaconda.
- Python 3.8 version is used for our work and have used the Tkinter library. Tkinter library is built in Python and generally used for creating GUI applications which is used for creation of GuidoBot.
- For clustering the user inputs to match with the predefined queries to the database we use a similarity technique named as cosine similarity. In this cosine similarity [19], we are maintaining the threshold point as 0.7 in order to match the similarity. Between the user input queries to the predefined queries.

B. JSP & JDBC
- Java Server Pages (JSP) it is a server-side technology which we used for creating web applications, and to create dynamic web content and we also used JDBC which acts as a bridge between middleware and backend databases(MySQL).

C. MySQL
- As a backend database for the web application we used MySQL database which is used for creating tables and storing data.

VI. EXPERIMENTAL SET UP AND RESULTS

A. Dataset

We have collected the data from the website where the faculty updates the data and from the database which is already available in the MySQL server.

B. Procedure

Step 1: User (Faculty) directly sign in with registered credentials in Guido chatbot.

Step 2: After validating registered credentials user (Faculty) will be redirected to the interface of the chatbot which displays three results, they are Notifications, Chat, and Task Findings.

Step 3: When faculty opens the Notifications tab then he/she is having different types of notifications that is pending tasks/goals, assigned tasks, change of Task Findings.

Step 4: When faculty opens the Chat tab then he/she can perform task updates, can change his/her goal progress, and manage their profile updates.

Step 5: When faculty opens the Task Findings tab then he/she can view his/her pending tasks and their deadlines, can update his task status using GuidoBot.

C. Results
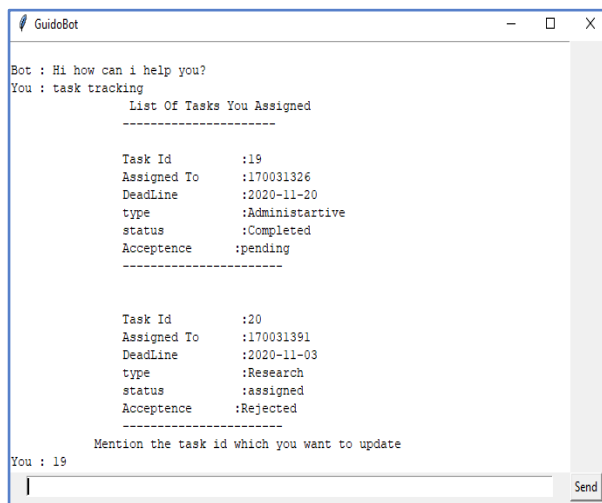
Fig 5 Description: The above figure reveals that assigner can track the task that he assigned, initially chatbot displays all the tasks that are assigned by him.
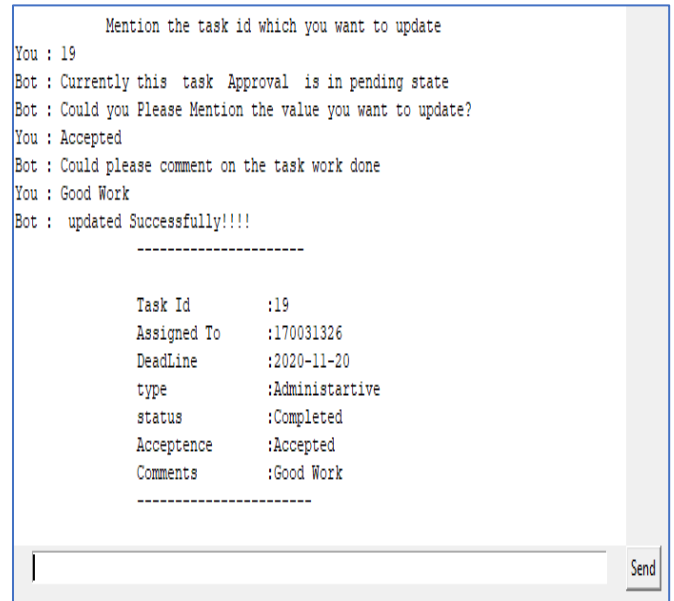
Fig 6 Description: The above figure represents that the assigner can track the task by mentioning the task id and he can accept or reject the task. User can also mention his comments regarding the work.
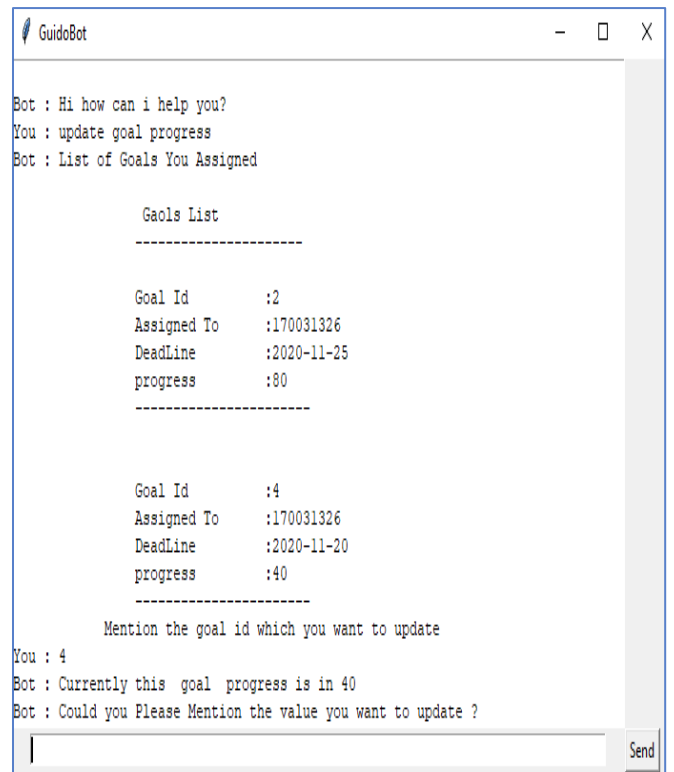

Fig 7 Description: The above figure represents how to interact with the chatbot to update the goal progress. Chatbot displays the list of goals assigned by the assigner. With the help of Goal id assigner can update the goal progress.

Fig 8 Description: The above figure represents the initial progress of the goal in the chatbot. Assigner can increment it with the percentage of completion as a progress.

Fig 9 Description: The above picture represents Goal Tracking page of Assigner, where he/she can view complete data along with deadline and status of goals that he/she assigned and he/she can able to change the progress of goals based on work done by assignee.

Fig 10 Description: The above picture represents the Task Tracking page of Assigner where he can track all the tasks that he assigned along with their deadlines and able to check their status and Approves the tasks.

VII. CONCLUSION

Technology as a concept is continuously working in providing new ways to save our time, and effort. In this research we are providing a chatbot assistance for the application Faculty Goal Tracking system. The Guido bot is like having a personal assistant that exists in your phone or computer. Instead of someone else running your schedule user can do it on his own. Faculty can utilize this Guido bot to update their profile, manage their goals and tasks in the goal tracking application in the best possible manner with the timely updates and notifications.

VIII. REFERENCES

[1] Sreelakshmi, A., Abhinaya, S., Nair, A., & Jaya Nirmala, S. (2019). A question answering and quiz generation chatbot for education. 2019 Grace Hopper Celebration India (GHCI).

[2] Shinde, M. S., Nahar, M. S., & Shelke, S. (2019). Intelligent chatbot for guided navigation of repository contend. IJARCCE, 8(5).

[3] Sandu, N., & Gide, E. (2019). Adoption of AI-chatbots to enhance student learning experience in higher education in India. 2019 18th International Conference on Information Technology Based Higher Education and Training (ITHET).

[4] Selvi, V., Saranya, S., Chidida, K., & Abarna, R. (2019). Chatbot and bullyfree chat. 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN).

[5] Sivcevic, D., Kosanin, I., Nedeljkovic, S., Nikolic, V., Kuk, K., & Nogo, S. (2020). Possibilities of used intelligence based agents in instant messaging on e-government services. 2020 19th International Symposium INFOTEH-JAHORINA (INFOTEH).

[6] Pandian, A. P. (2019). ARTIFICIAL INTELLIGENCE APPLICATION IN SMART WAREHOUSING ENVIRONMENT FOR AUTOMATED LOGISTICS. Journal of Artificial Intelligence, 1(02), 63- 72.

[7] Niranjan, P. Y., Rajpurohit, V. S., & Malgi, R. (2019). A survey on chat-bot system for agriculture domain. 2019 1st International Conference on Advances in Information Technology (ICAIT).

[8] Oh, K., Lee, D., Ko, B., & Choi, H. (2017). 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).

[9] Punjabi, S., Scthuram, V., Ramachandran, V., Boddu, R., & Ravi, S. (2019). Chat bot using API : Human to machine conversation. 2019 Global Conference for Advancement in Technology (GCAT).

[10] Chandy, A. (2019). A REVIEW ON IOT BASED MEDICAL IMAGING TECHNOLOGY FOR HEALTHCARE APPLICATIONS. Journal of Innovative Image Processing (JIIP), 1(01), 51-60.

[11] Akhila, G., Madhubhavana, N., Ramareddy, N. V., Hurshitha, M., & Ravinder, N. (2018). A survey on health prediction using human activity patterns through smart devices. International Journal of Engineering and Technology(UAE), 7(1), 226-229.

[12] Balram, G., & Kiran Kumar, K. (2018). Smart farming: Disease detection in crops. International Journal of Engineering and Technology(UAE), 7(2), 33-36.

[13] Buvanesvari, M., Uthayakumar, J., & Amudhavel, J. (2017). Fuzzy based clustering to maximize network lifetime in wireless mobile sensor networks. Journal of Advanced Research in Dynamical and Control Systems, 9(Special Issue 12), 2133-2143.

[14] Umar, S., Subbarayudu, Y., Kumar, K. K., & Bashwanth, N. (2017). Designing of dynamic re-clustering leach protocol for calculating total residual time and performance. International Journal of Electrical and Computer Engineering, 7(3), 1286-1292.

[15] Vijay Bhaskar Reddy, Y., Reddy, L. S. S., & Sai Satya Naryana Reddy, S. (2017). Comparative study of density-based clustering algorithms. International Journal of Civil Engineering and Technology, 8(12), 763-767.

[16] Kanaka Durga, K., & Rama Krishna, V. (2016). Automatic detection of illegitimate websites with mutual clustering. International Journal of Electrical and Computer Engineering, 6(3), 995-1001.

[17] Kousar Nikhath, A., & Subrahmanyam, K. (2016). Incremental evolutionary genetic algorithm based optimal document clustering (ODC). Journal of Theoretical and Applied Information Technology, 87(3), 479-488.

[18] Nirmal, K. R., & Satyanarayana, K. V. V. (2016). Issues of K means clustering while migrating to map reduce paradigm with big data: A survey. International Journal of Electrical and Computer Engineering, 6(6), 3047-3051.

[19] Durga Rajesh, K. V., Krishna Akhil, G., Teja Kumar, A., Vivek, K., & Radha Krishna, B. (2017). An effective similarity based sheep flock heredity algorithm to anticipate number of cells. Journal of Advanced Research in Dynamical and Control Systems, 9(Special issue 14), 2720-2726.

[20] Raman, D., Bezawada, B., Rajinikanth, T. V., & Sathyanarayan, S. (2017). Static program behavior tracing for program similarity quantification