COLLEGE ENQUIRY CHATBOT

A DESIGN PROJECT REPORT

Submitted by

SARASWATHY P L

SHANMATHI S

UMA MAHESWARI K

VEDHA VARSHINI V

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

NOVEMBER, 2024

COLLEGE ENQUIRY CHATBOT

A DESIGN PROJECT REPORT

Submitted by

SARASWATHY P L (811722104135)

SHANMATHI S (811722104141)

UMA MAHESWARI K (811722104170)

VEDHA VARSHINI V (811722104178)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

NOVEMBER, 2024

K RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report titled **"COLLEGE ENQUIRY CHATBOT"** is bonafide work of the students **SARASWATHY P L(811722104135), SHANMATHI S(811722104141), UMA MAHESWARI K(811722104170), VEDHA VARSHINI V(811722104178)** who carried out the project under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
Dr.A.Delphin Carolina Rani M.E.,Ph.D.,
HEAD OF THE DEPARTMENT
PROFESSOR
Department of CSE
K Ramakrishnan College of Technology
(Autonomous)
Samayapuram – 621 112

SIGNATURE
Mr. M Saravanan M.E.,
SUPERVISOR
Assistant Professor Department of CSE
K Ramakrishnan College of Technology
(Autonomous)
Samayapuram – 621 112

Submitted for the viva-voice examination held on ……

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

We jointly declare that the project report on **"COLLEGE ENQUIRY CHATBOT"** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of Bachelor Of Engineering. This project report is submitted on the partial fulfilment of the requirement of the award of Degree of Bachelor Of Engineering.

Signature

SARASWATHY P L

SHANMATHI S

UMA MAHESWARI K

VEDHA VARSHINI V

Place: Samayapuram

Date:

ACKNOWLEDGEMENT

It is with great pride that we express our gratitude and indebtness to our institution **"K RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)"**, for providing us with the opportunity to do this project.

We are glad to credit honorable chairman **Dr. K RAMAKRISHNAN, B.E.,** for having provided for the facilities during the course of our study in college.

We would like to express our sincere thanks to our beloved Executive Director **Dr. S KUPPUSAMY, MBA, Ph.D.,** for forwarding our project and offering adequate duration to complete it.

We would like to thank **Dr. N VASUDEVAN, M. Tech., Ph.D.,** Principal, who gave opportunity to frame the project with full satisfaction.

We whole heartily thank **Dr. A DELPHIN CAROLINA RANI, M.E., Ph.D.,** Head of the Department, **COMPUTER SCIENCE AND ENGINEERING** for providing her support to pursue this project.

We express our deep and sincere gratitude and thanks to our project supervisor **Mr. M. SARAVANAN, M.E.,** Department of **COMPUTER SCIENCE AND ENGINEERING**, for his incalculable suggestions, creativity, assistance and patience which motivated us to carry out this project.

We render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course. We wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

ABSTRACT

A chatbot, usually referred to as a chatterbot, attempts to have a conversation with a person. When a question is posed, the system has the ability to detect sentences and select the proper answer. The response principle is the matching of the user's input phrase. The current technical project involves building a professional system for a college help desk employing an android-based chatbot, artificial intelligence technology, and virtual assistance (human- machine communication), then sending that natural language to a server. Chatbot systems have become increasingly popular for automating interactions with users and providing information in various domains, including college enquiries. In this paper, we propose a chatbot system for college enquiry using a knowledgeable database. The system utilizes a knowledgeable database that contains relevant information about the college, such as courses, faculty, campus facilities, and admissions procedures. The system employs various algorithms, including rule-based, retrieval-based, natural language processing (NLP), and machine learning algorithms, to understand and respond to user queries in a context-aware manner. The rule- based algorithms provide predefined rules and patterns for handling specific intents or frequently asked questions, while the retrieval-based algorithms search the knowledgeable database for relevant information.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

S.No	ABBREVIATIONS	EXPANSION
1	AI	Artificial Intelligence
2	CNN	Convolution Neural Network
3	GUI	Graphical User Interface
4	LSTM	Long Short Term Memory
5	NER	Named Entity Recognition
6	NLG	Natural Language Generation
7	NLP	Natural Language Processing
8	NLTK	Natural Language Tool Kit
9	UML	Unified Modeling Language

CHAPTER – 1

INTRODUCTION

1.1 GENERAL INFORMATION:

This Application is for college students, staff, and parents. Easy way to interaction and time consuming. This project is mainly targeted at colleges and the synchronization of all the sparse and diverse information regarding regular college schedule. Generally, students face problems in getting correct notifications at the correct time, sometimes important notices such as campus interview, training and placement events, holidays, and special announcements. Smart Campus tries to bridge this gap between students, teachers, and college administrators. Therefore in the real world scenario, such as college campus, the information in the form of notices, oral communication, can be directly communicated through the android devices and can be made available for the students, teachers directly for their android devices and the maintenance of application will be easier in later future because of the use of architectural MVC which separates the major works in the development of an application such as data management, mobile user interface display and web service which will be the controller to make sure for fast and efficient maintenance of application.

The College bot project is built using artificial algorithms that analyses user's queries and understand user's message. This System is a web application which provides answer to the query of the student. Students just must query through the bot which is used for chatting. Students can chat using any format there is no specific format the user has to follow. The System uses built in artificial intelligence to answer the query. The answers are appropriate what the user queries. The User can query any college related activities through the system. The user does not have to personally go to the college for enquiry. The System analyses the question and then answers to the user.

The system answers to the query as if it is answered by the person. With the help of artificial intelligence, the system answers the query asked by the students.The system replies using an effective Graphical user interface which implies that as if a real person is talking to the user. The user just must register himself to the system and has to login to the system.

Based on the recent epidemiological situation, the increasing demand and relianceon electronic education has become very difficult to access to the university due to the curfew imposed, and this has led to limited access to information for academics at the university.

This project aims to build a chatbot for Admission and Registrationto answer every person who asks about the university, colleges, majors, and admission policy. Artificial intelligence (AI) is a branch of computer science that focuses on creating machines that can perform tasks that typically require human intelligence, such as perception, reasoning, learning, and decision-making.

Natural Language Processing: AI algorithms enable chatbots to understand natural language inputs from users and interpret them accurately.

Machine Learning: AI algorithms enable chatbots to learn from user interactions and improve their responses over time. Machine learning algorithms analyze the data collected from user interactions and identify patterns and trends. Based on this analysis, the chatbot can be trained to provide more accurate and relevant responses.

Personalization: AI algorithms enable chatbots to personalize their responses based on user preferences and behavior. By analyzing user data, chatbots can tailortheir responses to each user's specific needs and preferences.

1.2 PROBLEM STATEMENT

At the start of each academic semester, registration opens for those wishing to jointhe university in various disciplines, and telephone calls for admission and registration abound. This leads to an increase in the loads and work for the employees of the Deanship of Admission and Registration as a result of the constantpressure of those wishing to register and their families by flocking to the Deanship, so the employees are not able to answer the phone calls and social media. This often leads to many students who wish to register to be ignored.

The process of providing information and support to prospective and current students in a timely and efficient manner is a challenge for colleges, leading to frustration and dissatisfaction among users.

Knowledge database that contains all relevant information about the college and itsoperations. This database should be regularly updated to ensure that the information provided by the chatbot is accurate.

1.3 OBJECTIVES

- Save effort and time for both the admission and registration staff and students who wish to enroll.
- Provide detailed information about colleges and majors.
- Easy access to information.
- To minimize the time required to solve the queries.
- To give response to the user based on queries.
- To simplify communication between user and machine.

1.4 SYSTEM ARCHITECTURE

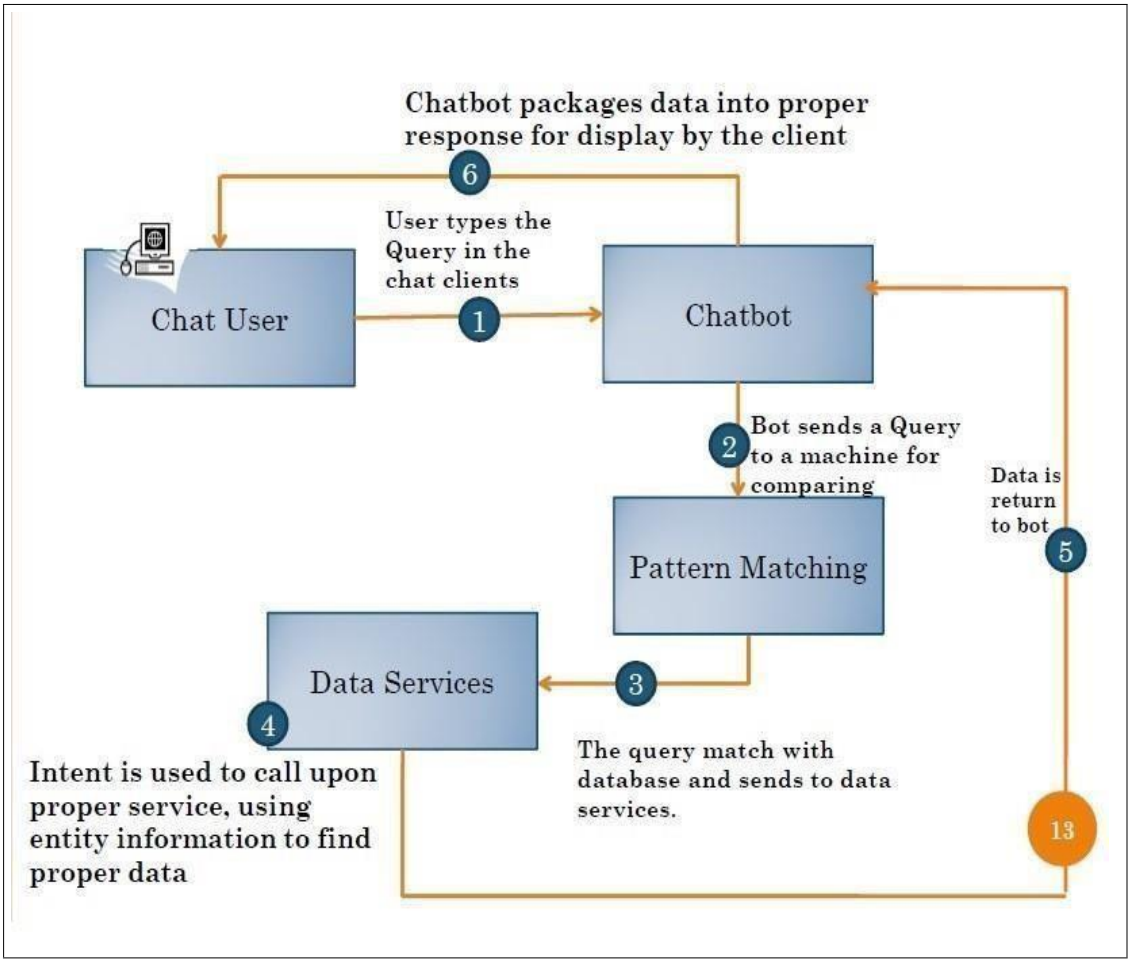

Figure 1.1 System Architecture

A block diagram is a diagram of a system in which the principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks. It may also show how the system operates, what are its inputs and outputs at various stages, and how the information, and/or materials flow through it. The block diagram for" Online chatting system for college enquiry knowledgeable Database" The proposed system has a client server architecture. All the information will be kept in an optimized database on the central server. This information can be accessed by the users through the android application installed on their smartphones (client machines). Each client machine will have an improved user interface.

Modules Client-Server (chat user): The proposed system has a client server architecture. All the information will be kept in an optimized database on the central server. This information can be accessed by the users through the android application installed on their smartphones (client machines). Each client machine will have an improved user interface.

Chatbot: A chatbot is a technology that allows users to have natural conversations to access content and services. Chatbots typically take the form of a chat client, leveraging natural language processing to conduct a conversation with the user. Chatbots control conversation flow based on the context of theusers requests and respond with natural language phrases to provide direct answers, request additional information or recommend actions that can be taken.

Pattern matching: Bot send a query to a machine for comparing. The query match with database sends to data services.

Data Services: Intent is used to call upon proper service. using entity information to find proper data. Hence all the modules described above are completed in polynomial time sec t, So the problem is P.

1.5 STATEMENT SCOPE

In today's world as there are everything is digital. In education system work is very lengthy and time consuming and required extra manpower. We develop this application for students, teachers, parents, and guest. In this project we implement android application due to this application The Student does not have to go personally to college office for the enquiry.

The application enables the students to be updated with college cultural activities. If application saves time for the student as well as teaching and non-teaching staffs. It is useful for parents also to show his/her child marks and important notices.

1.6 NATURAL LANGUAGE PROCESSING

NLP is an interdisciplinary subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, how to program computers to process and analyze large amounts of natural language data. The goal is a computer capable of "understanding" the contents of documents, including the contextual nuances of the language within them. The technology can then accurately extract information and insights contained in the documents as well as categorize and organize the documents themselves. NLP techniques allow the chatbot to understand the natural language queries of users and provide accurate and relevant responses. NLP is a critical component of many applications that involve language, such as chatbots, voice assistants, machine translation, sentiment analysis, and more. It involves several techniques and approaches, including statistical modeling, machine learning, deep learning, and rule- based systems, to analyze and process natural language data. NLP is a rapidly evolving field, and recent advances in machine learning and deep learning have led to significant improvements in the accuracy and performance of NLP models. Challenges in natural language processing frequently involve speech recognition, natural-language understanding, and natural-language generation.

In early days, many language-processing systems were designed by symbolic methods, i.e., the hand-coding of a set of rules, coupled with a dictionary lookup such as by writing grammars or devising heuristic rules for stemming.

More recent systems based on machine-learning algorithms have many advantages over hand-produced rules:

- The learning procedures used during machine learning automatically focus on the most common cases, whereas when writing rules by hand it is often not at all obvious where the effort should be directed.

- Automatic learning procedures can make use of statistical inference algorithms to produce models that are robust to unfamiliar input (e.g. containing words or structures that have not been seen before) and to erroneous input (e.g. with misspelled words or words accidentally omitted). Generally, handling such input gracefully with handwritten rules, or, more generally, creating systems of handwritten rules that make soft decisions, is extremely difficult, error-prone and time- consuming.

- Systems based on automatically learning the rules can be made more accurate simply by supplying more input data. However, systems based on handwritten rules can only be made more accurate by increasing the complexity of the rules, which is a much more difficult task. In particular, there is a limit to the complexity of systems based on handwritten rules, beyond which the systems become more and more unmanageable. However, creating more data to input to machine- learning systems simply require a corresponding increase in the number of man-hours worked, generally without significant increases in the complexity of the annotation process.

Despite the popularity of machine learning in NLP research, symbolic methods are still (2020) commonly used:

- When the amount of training data is insufficient to successfully apply machine learning methods, e.g., for the machine translation of low- resource languages such as provided by the Apterium system,

- For pre-processing in NLP pipelines, e.g., tokenization, or

- For postprocessing and transforming the output of NLP pipelines, e.g., for knowledge extraction from syntactic parses.

Sentiment analysis: Sentiment analysis is the process of identifying the emotion or sentiment behind a user's query. This can be useful in chatbots designed to provide emotional support or mental health services. Sentiment analysis is typically done using machine learning algorithms that are trained on large datasets of text labeled with sentiment.

Text generation: Text generation techniques can be used to generate natural language responses to user queries. These techniques use machine learning algorithms to analyze the context of the user's query and generate a relevant response. Text generation can be particularly useful in chatbots designed to handle complex or multi-turn conversations.
Natural Language Processing (NLP) techniques can play a crucial role in developing a college enquiry chatbot.

Intent recognition: NLP techniques can be used to recognize the intent of user queries, allowing the chatbot to provide appropriate responses.

Named entity recognition (NER): NER is a subtask of entity recognition that focuses specifically on identifying named entities such as people, organizations, andlocations. NER is commonly used in chatbots designed for customer service or support, where identifying customer names or order numbers is important.

Language translation: Chatbots can be designed to provide multilingual support by using language translation techniques. Machine learning algorithms are trained on large datasets of text in multiple languages to provide accurate translations.

Overall, NLP techniques are critical to the success of chatbots, allowing them to accurately understand user queries and provide relevant and personalized responses.

CHAPTER – 2

LITERATURE SURVEY

Professor Girish Wadhwa suggested that the institution build an inquiry chatbot using artificial intelligence in March-April 2017. Algorithms that might analyze consumer inquiries and recognize consumer messages. This machine might be a chatbot with the intention to provide solutions to students' questions. Students actually need to pick out a category for department requests and then request a botto be used for chat. The project's main goal is to develop an algorithm that may be used to correct the answers to queries that customers ask. It is essential to create a database where all related statistics can be kept as well as to expand the online interface. A database can develop to be able to compile information on queries, responses, key words, logs, and messages. 2016 saw Bayu Setiaji publish "Chatbot the usage of database knowledge." A chatbot is made to communicate with technology.

Machine learning is built to recognize sentences and concluded, such as the answer to a question. Personalized message, i.e. A request is saved in accordance with theresponse. The more similarly the statements are stated, the more it will be marked as similarity of the sentences. It is then answered in light of the answers from the first sentence. The sentence similarity calculator breaks the input sentence down into its component letters. A database stores the knowledge of chatbots. A chatbot has interfaces, and the database control system's access point through this interface is at its core. The Chatbot application was created using a variety of programming languages with the addition of a user interface that allows users to give input and get a response. Starting with the symbol of entity date, which produced 11 entities and their cardinalities, the structure and building of tables was done as an indication of the knowledge contained inside the database. SQL was used in a way that was tailored to the model that was kept inside the programme.

Scripts was presented by Md. Shahriare Satu and Shamim-Ai- Mamun. They asserted that entirely AIML-based chatbots are easy to set up, lightweight, and eco- friendly to use. post provides information on the various ways that chatbots are used. An AIML and LSA based chatbot was created by ThomasN. T. and Amrita Vishwa to provide customer support on e-commerce platforms.

2.1 OPEN PROBLEMS IN EXISTING SYSTEM

There are several open problems that need to be addressed in college enquiry chatbots to improve their performance and provide better user experience. Here aresome of the key open problems in college enquiry chatbots:

Intent Identification: One of the primary challenges in developing a college enquirychatbot is accurately identifying the user's intent. College enquiries can cover a wide range of topics, and the chatbot needs to correctly identify the user's intention to provide an appropriate response.

Accuracy: While chatbots can provide quick and convenient access to information, they are not always accurate in their responses. This is because the chatbot's database may not always be up-to-date, or the natural language processing algorithms may not be able to correctly interpret the user's queries.

Knowledge Base Management: A college enquiry chatbot needs to have access to a large amount of information about the college, including admission criteria, course offerings, faculty, and campus facilities. Managing this knowledge base is a significant challenge, as the information is often dispersed across multiple sources and needs to be kept up-to-date.

Natural Language Processing: Chatbots need to be able to understand and process natural language inputs accurately. However, natural language processing (NLP) technology is still in its early stages, and there are many challenges in accurately interpreting the meaning of user queries.

Chatbots can be integrated with a variety of channels, including websites, social media platforms, and messaging apps, to provide convenient access to information. Natural language processing (NLP) and machine learning (ML) techniques are commonly used in college enquiry chatbots to understand user queries and providerelevant responses. The use of chatbots in college enquiry systems can help to reduce workload for administrative staff and free up time for more complex tasks.

Personalization is an important feature of successful college enquiry chatbots, and can be achieved through the use of user data and machine learning techniques. Chatbots can help colleges to collect valuable data on user behaviour and preferences, which can be used to improve the quality of the chatbot's responses.

Several studies have focused on designing chatbots that can simulate human- like conversations and provide a personalized experience to users.

Multilingual Support: Colleges often have students from different parts of the world, speaking different languages. Providing multilingual support in college enquiry chatbots is a challenge that requires advanced NLP capabilities and a well-designed language model.

Personalization: To provide a better user experience, college enquiry chatbots need to personalize their responses based on the user's profile, preferences, and history. This requires advanced machine learning algorithms that can analyze userdata and provide tailored responses.

Context Management: College enquiries often involve complex and multi- turn conversations. Chatbots need to be able to maintain context across these conversations to provide accurate and relevant responses.

User Engagement: Finally, chatbots need to be engaging and interactive to keep users interested and motivated to continue using them. This requires designing chatbots that can simulate human-like conversations and provide relevant and interesting information to users.

2.2 INFERENCES FROM LITERATURE SURVEY

Based on a literature survey of college enquiry chatbots, several key inferences canbe drawn:

College enquiry chatbots are becoming increasingly popular: There is a growing trend of colleges and universities adopting chatbots to handle student enquiries. Several studies have shown that chatbots can significantly reduce the workload on college administrators and provide faster, more efficient.

Natural language processing (NLP) is a critical component of college enquiry chatbots: NLP technology is used to understand and interpret user queries, and to generate natural language responses. Several studies have focused on improving the accuracy and effectiveness of NLP in college enquiry chatbots.

Machine learning (ML) algorithms are being used to improve the performance of college enquiry chatbots: ML algorithms are used to train chatbots on large datasets of student queries and responses. This helps chatbots to learn from past interactions and provide more accurate and relevant responses.

Chatbots are being used to support a wide range of college enquiries: College enquiry chatbots can handle a wide range of enquiries, including admission inquiries, course registration, financial aid, campus facilities, and career services. The success of college enquiry chatbots depends on effective design and development, including careful consideration of user needs, the use of appropriate testing and optimization.

Multilingual support is a growing area of research in college enquiry chatbots: Many colleges and universities have a diverse student population, and providing multilingual support is essential to ensure that all students can access the information and services they need.

Context management is a critical challenge in college enquiry chatbots: College enquiries often involve complex and multi-turn conversations. Chatbots need to be able to maintain context across these conversations to provide accurateand relevant responses.

User experience is a critical factor in the success of college enquiry chatbots: Chatbots need to be engaging, interactive, and easy to use to keep students interested and motivated to use them. Several studies have focused on designing chatbots that can simulate human-like conversations and provide a personalized experience to users. College enquiry chatbots can help colleges to provide more efficient and effective customer service to prospective and current students, and canimprove the overall user experience.

Chatbots can be integrated with a variety of channels, including websites, social media platforms, and messaging apps, to provide convenient access to information. Natural language processing (NLP) and machine learning (ML) techniques are commonly used in college enquiry chatbots to understand user queries and providerelevant responses. The use of chatbots in college enquiry systems can help to reduce workload for administrative staff and free up time for more complex tasks.

Personalization is an important feature of successful college enquiry chatbots, and can be achieved through the use of user data and machine learning techniques. Chatbots can help colleges to collect valuable data on user behaviour and preferences, which can be used to improve the quality of the chatbot's responses.

CHAPTER - 3

REQUIREMENT ANALYSIS

3.1 SOFTWARE AND HARDWARE REQUIREMENTS SPECIFICATION

SOFTWARE AND HARDWARE REQUIREMENTS:

Hardware:

Operating system	: Windows 7 or 7+
RAM	: 2 GB MEMORY
Hard disc or SSD	: More than 500 GB
Processor	: Processor Dual Core

Software:

Software's	: Python 3.6 or high version
IDLE	: PyCharm.
Framework	: Flask

3.2 SYSTEM USE CASE

A college enquiry chatbot can have several use cases, including:

Admission Enquiry: The chatbot can provide information about the admissionprocess, eligibility criteria, important dates, and documents required for admission.

Course Information: The chatbot can provide detailed information about the courses offered by the college, including the duration of the course, syllabus, fees, and career opportunities.

Campus Facilities: The chatbot can provide information about the various facilitiesavailable on the college campus, such as libraries, laboratories, sports facilities, andaccommodation options.

Fees and Scholarships: The chatbot can provide information about the fees structure for different courses and scholarships available for students based on theiracademic performance.

Important Dates: The chatbot can remind students about important dates such asadmission deadlines, fee payment dates, and exam schedules.

FAQs: The chatbot can answer frequently asked questions by students, such as how to apply for admission, how to check the admission status.

Student life: The chatbot can provide information about student life at the college, including clubs and societies, extracurricular activities, and student resources.

Counseling: The chatbot can provide counseling to students regarding their careeroptions, course selection, and academic performance.

Academic support: The chatbot can assist students with academic enquiries, including course registration, exam schedules, and study resources.

Admission and enrolment enquiries: The chatbot can assist prospective students with admission and enrolment enquiries, including deadlines, application requirements, and documentation. Overall, a college enquiry chatbot can provide aseamless and hassle-free experience for students who are looking for information about the college and its courses.

CHAPTER – 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 STUDY OF THE PROJECT

This project is mainly targeted at colleges and the synchronization of all the sparseand diverse information regarding regular college schedule. Generally, students face problems in getting correct notifications at the correct time, sometimes important notices such as campus interview, training and placement events, holidays and special announcements. Smart Campus tries to bridge this gap between students, teachers, and college administrators. Therefore in the real worldscenario, such as college campus, the information in the form of notices, oral communication, can be directly communicated through the android devices and can be made available for the students, teachers directly for their android devices and the maintenance of application will be easier in later future because of the use of architectural MVC which separates the major works in the development of an application such as data man agreement, mobile user interface display and web service which will be the controller to make sure for fast and efficient maintenance of application.

A study is carried out to select the best system that meets the performance requirements. Feasibility is the determination of whether a project is worth doing or not. The process followed in making this determination is called a feasibility study. This type of study determines if a project can and should be taken. Since the feasibility study may lead to the commitment of large resources, it becomes necessary that it should be conducted competently and that no fundamental errors of judgment are made. Depending on the results of the initial investigation, the survey is expanded to a more detailed feasibility study. Feasibility study is a test of system proposal according to its work- ability, impact on the organization, ability to meet user needs, and effective use of resources. The objective of the feasibility study is not to solve the problem but to acquire a sense of its scope. During the study, the problem definition is crystallized and aspects of the problem to be included in the system are determined.

Save timing of students and teachers and also save extra manpower. Student can see all document related college like, notice, study material, question papers etc. on time to time and from any place whether student is present in college or not. And also reduce the work of staff. It is proper communication in between staff and students.

Natural language processing algorithms: To interpret user queries and generate accurate responses.

Knowledgeable database: To store information about college programs, courses, and admission requirements.

Recommendation engine: To suggest suitable programs based on the user's interests and qualifications.

User interface: To provide a user-friendly and intuitive interface for users to interactwith the chatbot.

Data collection and processing: To gather and organize information about college programs, courses, and admission requirements.

Algorithm development: To develop natural language processing algorithms that can interpret user queries and generate accurate responses.

Database design and implementation: To design and implement a knowledgeable database that can store and retrieve information about college programs, courses, and admission requirements.

User interface design and implementation: To design and implement a user interface that is intuitive and user-friendly.

Testing and evaluation: To test the chatbot system for accuracy, usability, and performance.

4.2 EXISTING METHODOLOGY

Knowledge graph creation: The first step is to create a knowledge graph that contains all the relevant information about college programs, courses, and admission requirements. This can be done using existing ontologies or by manuallycurating the knowledge graph.

4.2.1. To develop the problem under consideration and justify feasibility usingconcept of knowledge canvas and IDEA matrix.

I	D	E	A
Increase	Drive	Educate	Accelerate
Improve	Deliver	Evaluate	Associate
Ignore	Decrease	Eliminate	Avoid

TABLE 4.1 – IDEA

Learning objective: 1. Project feasibility

- Project feasibility
- Find Knowledge gap
- Learn IDEA matrix
- Knowledge canvas

IDEA Matrix:

IDEA matrix is nothing but a matrix representation of characteristic requirement ofthe project.

The IDEA matrix of our project can be thus represented as:

I	D	E	A
Increase efficiency of Search Engine.	Drive a search Engine which is smart enough to be search relevant search.	Educate the human to how to search appropriate result	Accelerate speed of Searching result.
Improve relevant search result.	Deliver the exact result of search with help of Smart crawler.	Evaluate technical advancements of Society for its betterment.	Associate database with Inventory system.
Ignore irrelevant result.	Decrease visiting to unwanted link of our search result.	Eliminate large amount of processing efforts.	Avoid processing in maintaining daily records of the database.

TABLE 4.2 – IDEA MATRIX

Brief explanation about each characteristic:

Increase: In our project we are thus increase the use and operating efficiency of current search engine. We are increasing searching capacity of the relevant result. Improve: Improve the traditional search engine by making it smarter using technologies such as Smart Crawler.

Ignore: We are ignoring the irrelevant result of given searches. Our traditional search engine gives both results relevant and irrelevant searches among from themwe take relevant search using smart technologies like smart crawler.

Drive: Hereby we are driving a smart search engine against a traditional search engine which helps us reducing extra search efforts.

Deliver: We are delivering a quick and easy solution for the maintenance of database that needs to be updated on regular interval.

Decrease: The extra visit to unwanted result will be decreased by using Smart Crawler and profession login option also provided on the smart crawler.

Educate: We are trying to make the management authority and efficiency of searchengine aware of technical advancements around.

Evaluate: By considering the searching on internet reviews and requirements whichneeds to be satisfied given by the users we are evaluating the technology to be usedalong with algorithms needs to reduce efforts.

Eliminate: By implementation of smart crawler need for massive number of systemprocessing is eliminated which leads to efficiency.

Accelerate: Searching is done at much higher speed as there would be we are usingsmart technologies and algorithms so that it removes unwanted results.

Associate: Here we are associating or linking database with the inventory so that if the sites go below threshold level inventory must make required arrangements so that the sides should not be unavailable.

Avoid: If any irrelevant search result in updating database goes may lead to wrong search result in the system. This needs to be avoided. Hence an updating mechanism is added with help of smart crawler.

KNOWLEDGE CANVAS:

Knowledge canvas is a graphical representation of knowledge gap between any twocomponents of the project considered.

Knowledge canvas Diagram

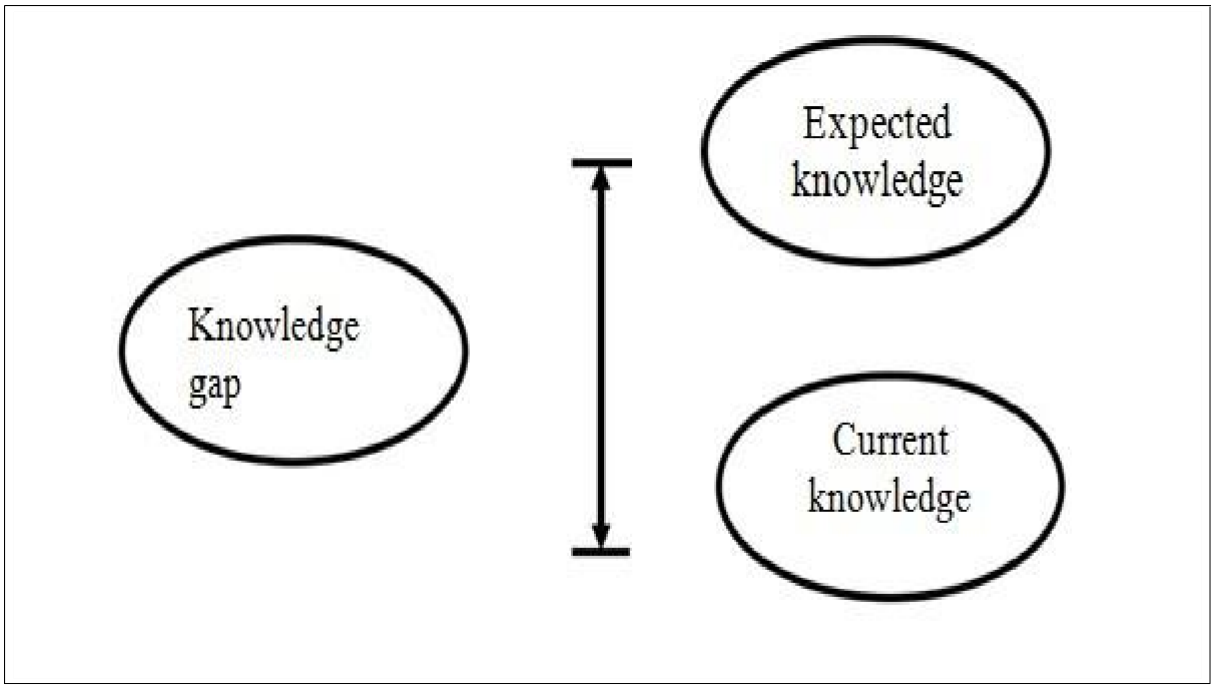

Fig 4.1 Knowledge Canvas Diagram

4.2.2. Project problem statement feasibility assessment using NP-Hard, NP-Complete.

P PROBLEM

Polynomial time solving. Problems which can be solved in polynomial time, which take time like O(n), O(n2), O(n3). E.g.: finding maximum element in an array or to check whether a string is palindrome or not.

So, there are many problems which can be solved in polynomial time.

NP PROBLEM

Non deterministic Polynomial time solving. An easy example of this is subset sum: given aset of numbers, does there exist subset whose sum is zero?.

But NP problems are checkable in polynomial time means that given a solution of a problem,we can check that whether the solution is correct or not polynomial.

NP-hard PROBLEM

If a problem is NP-hard, this means I can reduce any problem in NP to that problem. This means if I can solve that problem, I can easily solve any problem in NP. If we could solve an NP-hard problem in polynomial time, this would prove P = NP.

NP-complete PROBLEM

A problem is NP-complete if the problem is both NP-hard, and In NP.

Algorithms & Techniques:

Algorithm 1: Exact Pattern Matching AlgorithmO (N + K).

Complexity: It takes time to fetch URL from web-server, also to extract query entered by user. It takes data from database as well as from log file.

So, Time Complexity=o(n) OCR-Optical Character Recognition Complexity Analysis

Algorithm 2: OCR-Optical Character RecognitionO (N 2 log (N)).

Overall time required: O(N+K) +O (N 2 log(N))

Space Complexity:

More the storage of data more is the space complexity. Each time we store resultant data in log file also in database. We store URL (bookmarked) in database. So, moretime complexity.

4.3 PROPOSED METHODOLOGY

Admin:

Add Student: The Admin adds the student and the password is generated by thesystem and sent to the students Mail Id.

Add Course: The Admin is allowed to add the Course and its Subjects semesterwise.

Add Timetable: The Admin is allowed to add the timetable for the course semesterwise in the form of an .jpg

Add Schedule: The Admin is allowed to add the Schedule for the course semesterwise in the form of an .jpg

Add Booklet: The Admin adds the booklet limited to a pdf file only.

Add Test Solutions: The Admin adds the test solutions limited to a pdf file only.wise and out of 25.

Add College related information e.g., Events, workshop doc, photos, branch infowith photos. Which is useful for represent college.

Student:

Student Login: The Student is allowed to login into the App with password sent tohis/her email Id and is remembered once logged In.

View Booklet: The Student can see a list of the booklets limited to his/her courseand semester which are viewed by default by Google docs.

View College related information e.g., Events, workshop doc, photos, branch infowith photos. Which is useful for represent college.

Parent:

Parent Login: The Parent is allowed to login into the App with password sent tohis/her email Id and is remembered once logged In

4.4 PROJECT TASK SET/PROJECT MANAGEMENT PLAN:

- Task 1-Requirement Gathering, Review of papers
- Task 2-Defining problem statement
- Task 3-Identifying scope and requirements of project
- Task 4-Mathematical analysis
- Task 5-System design analysis
- Task 6-UML diagrams
- Task 7-System Implementation
- Task 8-System Testing
- Task 9-Result Analysis
- Task 10-Documentation

CHAPTER – 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT SETUP

A college enquiry chatbot can be built using a combination of LSTM (Long Short-Term Memory) and CNN (Convolutional Neural Network) models to process natural language inputs and generate appropriate responses.

Here is how it can work:

- **Data collection:** The first step is to collect a large amount of relevant data, such as frequently asked questions, course information, admission requirements, campus facilities, etc. This data will be used to train the chatbotmodel. The relevant data is taken from Concordia university for the overviewof the project.
- **Preprocessing:** The first step is to preprocess the text inputs to extract important features and remove any noise. This can involve steps such as tokenization, stemming, lemmatization, stop word removal, and spell correction.

Natural Language Processing is a subfield of data science that works with textual data. When it comes to handling the Human language, textual data is one of the most unstructured types of data available. NLP is a technique that operates behind the it, allowing for extensive text preparation prior to any output. Before using the data for analysis in any Machine Learning work, it's critical to analyse the data. To deal with NLP-based problems, a variety of libraries and algorithms are employed. For text cleaning, a regular expression(re) is the most often used library. The next libraries are NLTK (Natural language toolkit) and spacy, which are used to execute natural language tasks like eliminating stop words.

Pre-processing data is a difficult task. Text pre-processing is done in order to prepare the text data for model creation. It is the initial stage of any NLP project.

5.1.1. REMOVING STOP WORDS

To eliminate noise from data, data cleaning is essential in NLP. Stop words are the most frequently repeated words in a text that give no useful information. The NLTK library includes a list of terms that are considered stop words in English. [I, no, nor, me, mine, myself, some, such we, our, you'd, your, he, ours, ourselves, yours, yourself, yourselves, you, you're, you've, you'll, most, other] are only a few ofthem.

The NLTK library is a popular library for removing stop words, and it eliminates about 180 stop words. For certain difficulties, we can develop a customized set of stop words.

Using the add technique, we can easily add any new word to a collection of terms. Removing stop words refers to the process of considered to be common uninformative.

Removing stop words is a common preprocessing step in Natural Language Processing (NLP)
Stop words are words that do not carry much meaning in a sentence or text, such as:

- Articles (the, a, an)
- Conjunctions (and, but, or)
- Prepositions (in, on, at)
- Pronouns (I, you, he, she)
- Auxiliary verbs (is, are, am)

Removing stop words can help improve the performance of NLP models by:

- Improving model efficiency: By removing stop words, the model has to process less data will can improve its efficiency.

- Enhancing feature extraction: Removing stop words can help the model focus on the more important words in the text, which can improve feature extraction.

5.1.2. LEMMATIZATION

The process of reducing inflected forms of a word while verifying that the reduced form matches to the language is known as lemmatization. A lemma is a simplified version or base word. Lemmatization uses a pre-defined dictionary to saves word context and verify the word in the dictionary as it decreases. Organizes, organized, and organizing, for example, are all forms of organize. The lemma in this case is organize.The inflection of a word can be used to communicate grammatical categories such as tense (organized vs organize). Lemmatization is required since it aids in the reduction of a word's inflected forms into a particular element for analysis. It can also assist in text normalization and the avoidance of duplicate words with similar meanings.

5.1.3. LOWER CASING

When the text is in the same case, a computer can easily read the words since the machine treats lower and upper case differently. Words like Cat and cat, for example, are processed differently by machines. To prevent such issues, we must make the word in the same case, with lower case being the most preferable instance. In python lower () is a function that is mostly used to handle strings. The lower () function accepts no parameters. It converts each capital letter to lowercase to produce lowercased strings from the provided string. If the supplied string has no capital characters, it returns the exact string.

- **Intent Recognition:** The next step is to identify the intent behind the user's input. For example, if the user asks "What are the admission requirements for Computer Science?", the intent can be recognized as"Admission Requirements".

  This can be done using techniques such asrule-based systems, machine learning algorithms like Naive Bayes, or neural network models like LSTM.

- **Entity Recognition:** Once the intent is recognized, the chatbot needs to extract the relevant entities from the user's input. In the above example, the entities would be "Computer Science". This can be done using techniques such as Named Entity Recognition (NER) or Part-of-Speech (POS) tagging.

- **Dialogue Management:** The chatbot needs to maintain a conversation flowwith the user and respond appropriately to their inputs. This can be achieved using techniques such as rule-based systems, finite-state machines, orreinforcement learning algorithms.

- **Response Generation:** Finally, the chatbot generates a response to the user's input based on the intent and entities identified in the previous steps. The response can be a pre-defined template or a dynamically generated sentence. The response can be generated using techniques such as rule- based systems, templates, or machine learning algorithms like sequence-to-sequence models or Generative Pre-trained Transformer (GPT) models.

5.2 ALGORITHMS

5.2.1 Long Short-Term Memory (LSTM)

LSTM is a kind of recurrent neural network. In RNN output from the last step is fed as input in the current step. LSTM was designed by Hochreiter & Schmid Huber. It tackled the problem of long-term dependencies of RNN in which the RNN cannot predict the word stored in the long-term memory but can give more accurate predictions from the recent information. As the gap length increases RNN does not give an efficient performance.

LSTM can by default retain the information for a long period of time. It is used for processing, predicting, and classifying based on time- series data. Long Short- Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) that is specifically designed to handle sequential data, such as timeseries, speech, and text. LSTM networks can learn long-term dependencies in sequential data, which makes them well suited for tasks such as language translation, speech recognition, and time series forecasting. A traditional RNN has a single hidden state that is passed through time, which can make it difficult for the network to learn long-term dependencies. LSTMs address this problem by introducing a memory cell, which is a container that can hold information for an extended period.

The memory cell is controlled by three gates: the input gate, the forget gate, and the output gate. These gates decide what information to add to, remove from, and output from the memory cell. The input gate controls what information is added to the memory cell.

The forget gate controls what information is removed from the memory cell. And the output gate controls what information is output from the memory cell. This allows LSTM networks to selectively retain or discard information as it flows through the network, which allows them to learn long-term dependencies.

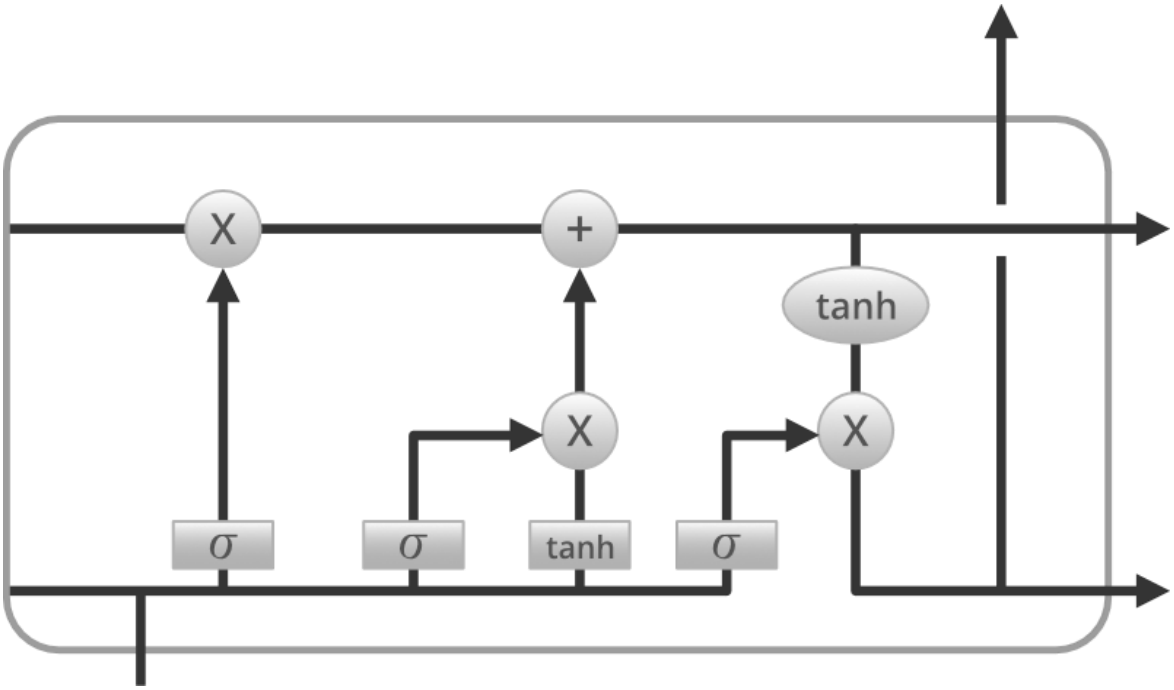

Fig 5.1 Structure of LSTM

LSTMs can be stacked to create deep LSTM networks, which can learn even more complex patterns in sequential data. LSTMs can also be used in combination with other neural network architectures, such as Convolutional Neural Networks (CNNs) for image and video analysis.LSTM has a chain structure that contains four neural networks and different memory blocks called cells. Information is retained by the cells and the memory manipulations are done by the gates.

There are three gates –

1. **Forget Gate:** The information that is no longer useful in the cell state is removed with the forget gate. Two inputs x_t (input at the particular time) and h_t-1 (previous cell output) are fed to the gate and multiplied with weight matrices followed by the addition of bias. The resultant is passed through an activation function which gives a binary output. If for a particular cell state the output is 0, the piece of information is forgotten and for output 1, the information is retained for future use.

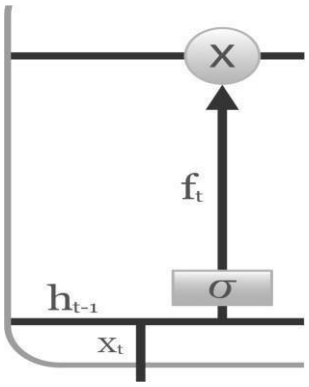

Fig 5.2 Forget Gate

2. **Input gate:** The addition of useful information to the cell state is done by the inputgate. First

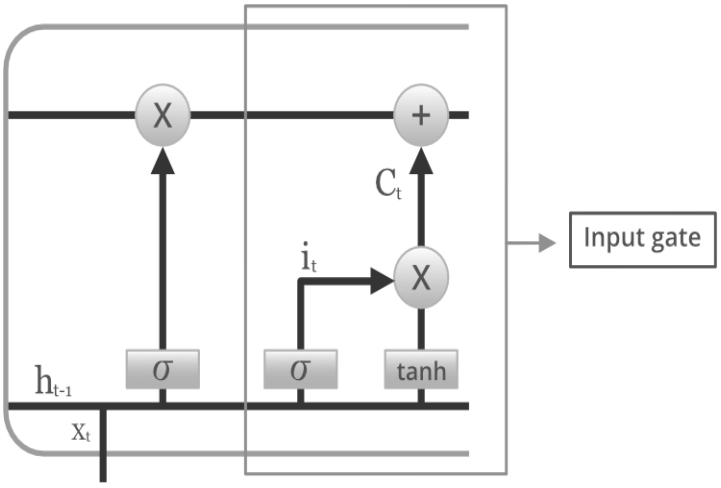

Fig 5.3 Input Gate

The information is regulated using the sigmoid function and filter the values to be remembered like the forget gate using inputs h_t-1 and x_t.

Then, a vector is created using tanh function that gives an output from -1 to +1, which contains all the possible values from h_t-1 and x_t. At last, the values of the vector and the regulated values are multiplied to obtain the useful information

3. **Output gate:** The task of extracting useful information from the current cell state to be presented as output is done by the output gate. First, a vector is generated by applying tanh function on the cell. Then, the information is regulated using the sigmoid function and filter by the values to be remembered using inputs h_t-1 and x_t. At last, the values of the vector and the regulated values are multiplied to be sent as an output and input to the next cell

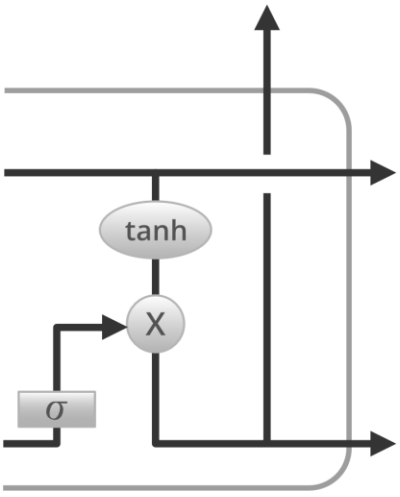

Fig 5.4 Output Gate

.

5.2.2 CONVOLUTIONAL NEURAL NETWORK (CNN)

A Convolutional Neural Network (CNN) is a type of deep learning algorithm commonly used in image recognition and computer vision applications. This stands for Convolution Neural Network where Image data is mapped to a target variable. They have proven to be successful in that they are now the techniques of choice forany form of prediction issue utilizing data as an input to the model. CNN is a multi- layered feed-forward neural network that is built by layering several hidden layers on top of one another in a certain sequence. These layers are frequently outlawed by several layers in CNN, while activation layers are usually enhanced by layers in the convolutional network. In the context of a college enquiry chatbot system, CNN can be useful in several ways:

Image Recognition: CNN can help the chatbot to identify images related to collegeenquiries. For example, if a user sends an image of a college campus, the chatbot can use a pre-trained CNN model to recognize the image and extract relevant information such as the name of the college, its location, and other details that canassist the user in their enquiry.

Data Analysis: CNN can be used to analyze textual data related to college enquiries. For example, if a user asks a question about admission requirements for a particular program, the chatbot can use a CNN model to extract the most importantkeywords and concepts from the text and provide a relevant response based on thatinformation.

Improved Accuracy: Using a CNN model can improve the accuracy of the chatbot'sresponses, as it can quickly and accurately analyze large amounts of data related to college enquiries and provide the most relevant responses to users.

Chatbot training: CNNs can be used as a part of the training process for chatbots. For example, CNNs can be used to analyze large datasets of user queries and responses to identify patterns and improve the chatbot's ability to understand and respond to user queries.

Text classification: CNNs can be used to classify user input into different categories or intents. This is useful in chatbots as it allows the chatbot to understand the user's query and respond appropriately. CNNs can learn to identify patterns in text data and can be trained on large datasets to improve their accuracy.

Entity extraction: CNNs can be used to extract relevant information from unstructured text data, such as course descriptions or faculty biographies. This canbe useful in chatbots for providing detailed information to users.

Contextual understanding: CNNs can be used to improve the chatbot's contextual understanding of user input.

Overall, CNN can be a valuable tool in a college enquiry chatbot system, as it can help to enhance the accuracy and effectiveness of the chatbot in responding to user enquiries, especially when it comes to analyzing visual and textual information. Contextual understanding refers to the ability to coin the context in which it is presented.

5.3 MODULE IMPLEMENTATION

5.3.1. RDFLIB

RDFLib is a pure Python package for working with RDF. RDFLib contains mostthings you need to work with RDF, including:

- Parsers and serializers for RDF/XML, N3, NTriples, N-Quads, Turtle, TriX,Trig and JSON-LD
- A Graph interface which can be backed by any one of a number of Storeimplementations
- Store implementations for in-memory, persistent on disk (Berkeley DB) andremote SPARQL endpoints
- A SPARQL 1.1 implementation - supporting SPARQL 1.1 Queries andUpdate statements
- SPARQL function extension mechanisms

5.3.2. RE

A RegEx, also known as a Regular Expression, is a string of characters that definesa search patterns. This module's functions allow to see if a given string matches a given regular expression.

5.3.3. RANDOM

The Python Random module is a built-in module for generating random integers in Python. These are sort of fake random numbers which do not possess true randomness. We can therefore use this module to generate random numbers, display a random item for a list or string, and so on.

5.3.4. CSV

The CSV module implements classes to read and write tabular data in CSV format. It allows programmers to say, “write this data in the format preferred by Excel,” or “read data from this file which was generated by Excel,” without knowing the precisedetails of the CSV format used by Excel. Programmers can also describe the CSV formats understood by other applications or define their own special- purpose CSV formats.

5.3.5. SPOTLIGHT

Data validation for Python, inspired by the Laravel framework. The main focus of the Spotlight library is on deep learning techniques such as matrix factorization, neural networks, and sequence modeling. It provides a flexible framework for building different types of recommendation systems, including collaborative filtering, content-based filtering, and hybrid models.

5.4 DATA FLOW DIAGRAMS

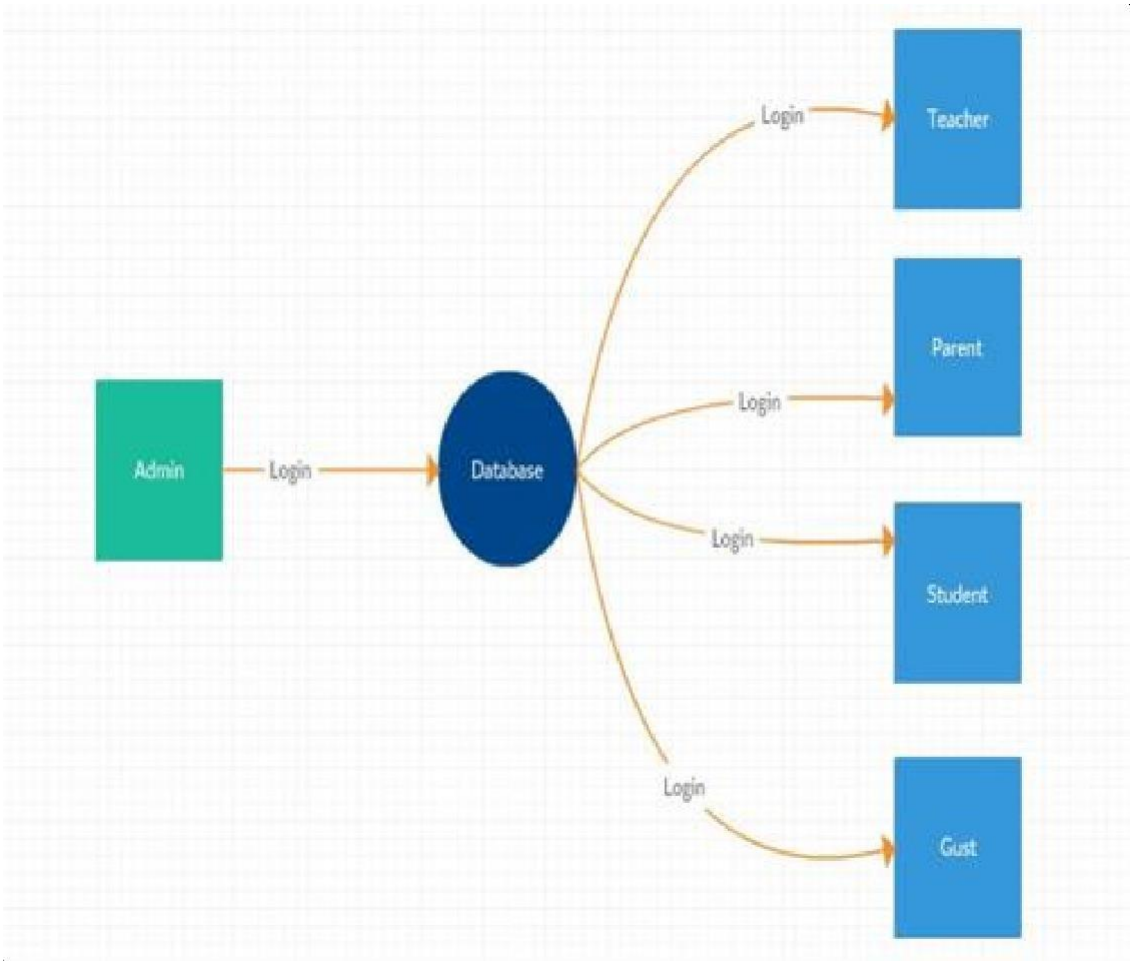

Fig 5.5.1 Level 0 Data Flow Diagram

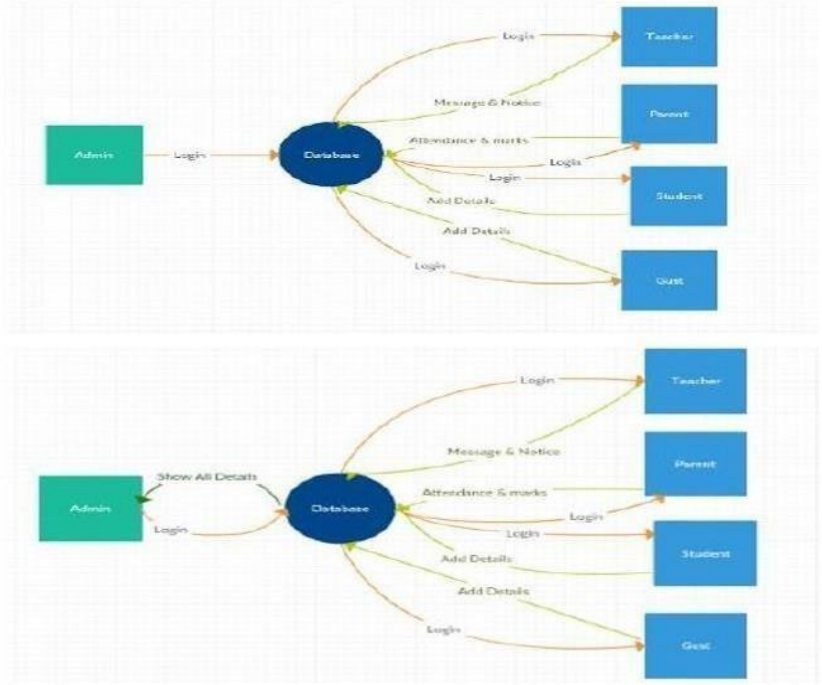

Fig 5.5.2 Level 1 Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the flow of data in a system. In the context of the chatbot system for college enquiry using a knowledgeable database, a DFD can be used to illustrate the flow of data between the various components of the system. The DFD can help in understanding the datainputs, processing, and outputs of the system. The DFD for the chatbot system can be divided into four main components: the user interface, the natural language processing engine, the knowledgeable database, and the response generation component. The user interface component receives the input queries from the user in natural language. The input query is then passed on to the natural language processing engine, which processes the query and extracts relevant information such as intent, entities, and sentiment.

5.5 USE CASE DIAGRAM

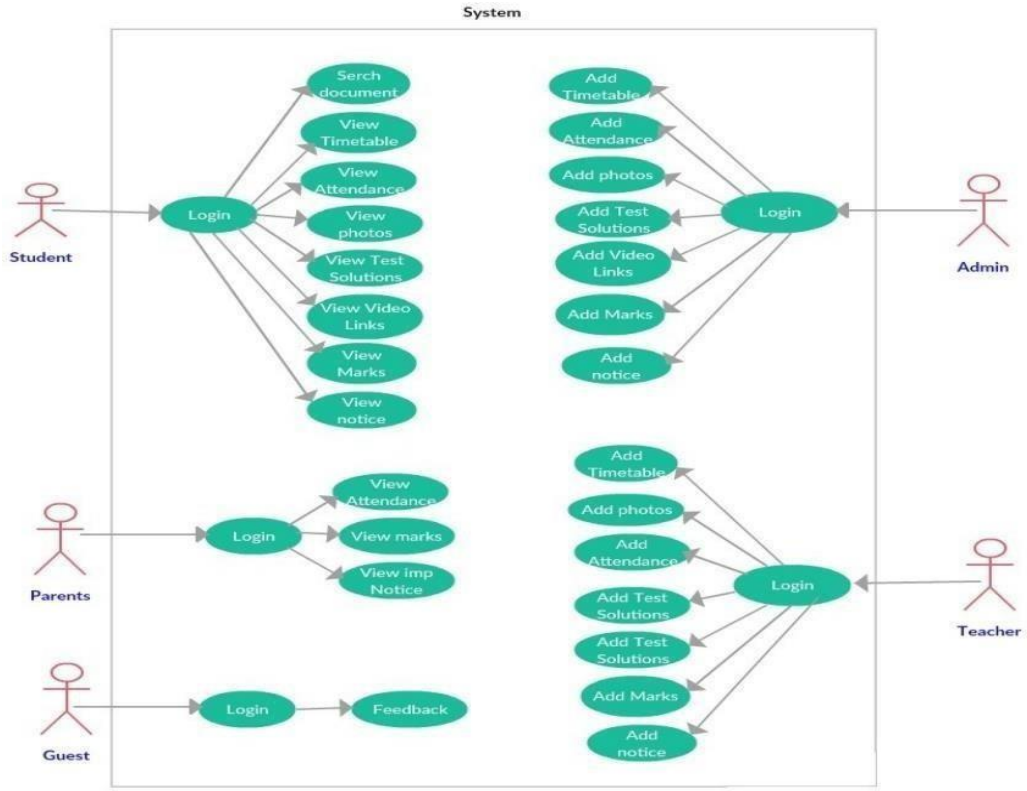

Fig 5.6 Use Case Diagram

A use case diagram is a graphical representation of the interactions between actors(users) and the system. In the context of the chatbot system for college enquiry using a knowledgeable database, a use case diagram can be used to identify the various use cases or scenarios in which the system is used.The use case diagram for the chatbot system can include the actors (users) such as prospective students, parents, and other stakeholders who are interested in obtaining information about the college. The various use cases can include querying information about courses,admission requirements, campus facilities, and other related information.

5.6 CLASS DIAGRAM

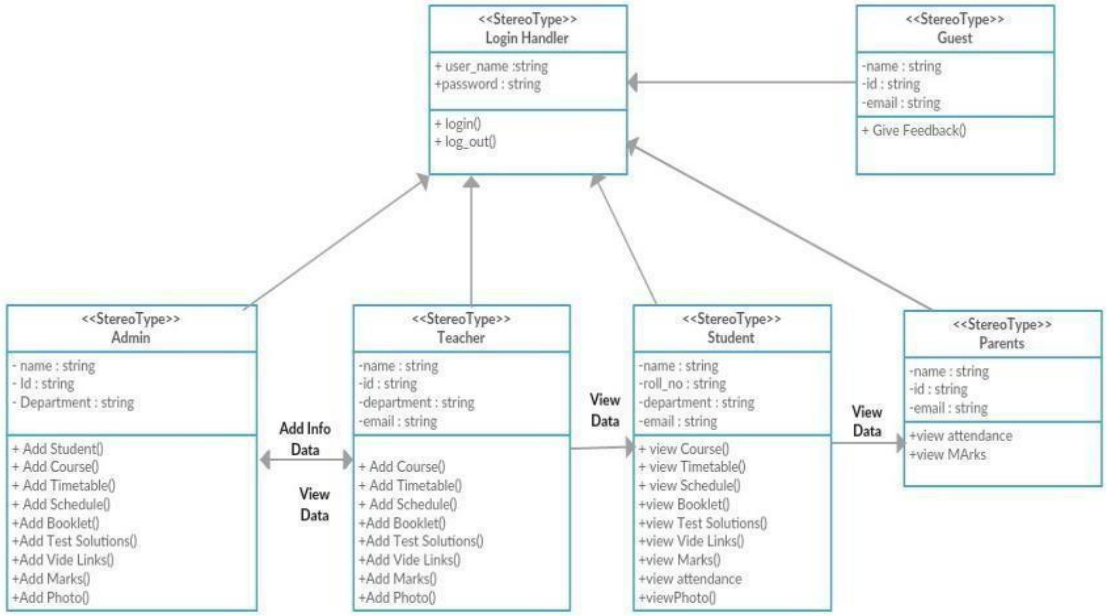

Fig 5.7 Class Diagram

A class diagram is a type of UML (Unified Modeling Language) diagram that represents the classes and their relationships in a system. In the context of the chatbot system for college enquiry using a knowledgeable database, a class diagram can be used to represent the various classes in the system and their relationships.The class diagram for the chatbot system can include classes such as User, Query, Response, Natural Language Processing Engine, Knowledgeable Database, Retrieval-based Algorithm, Rule-based Algorithm, Machine Learning Algorithm, Hybrid Approaches, and Feedback Mechanism. Each class can have attributes and methods that define its behavior and properties.

5.7 SEQUENCE DIAGRAM

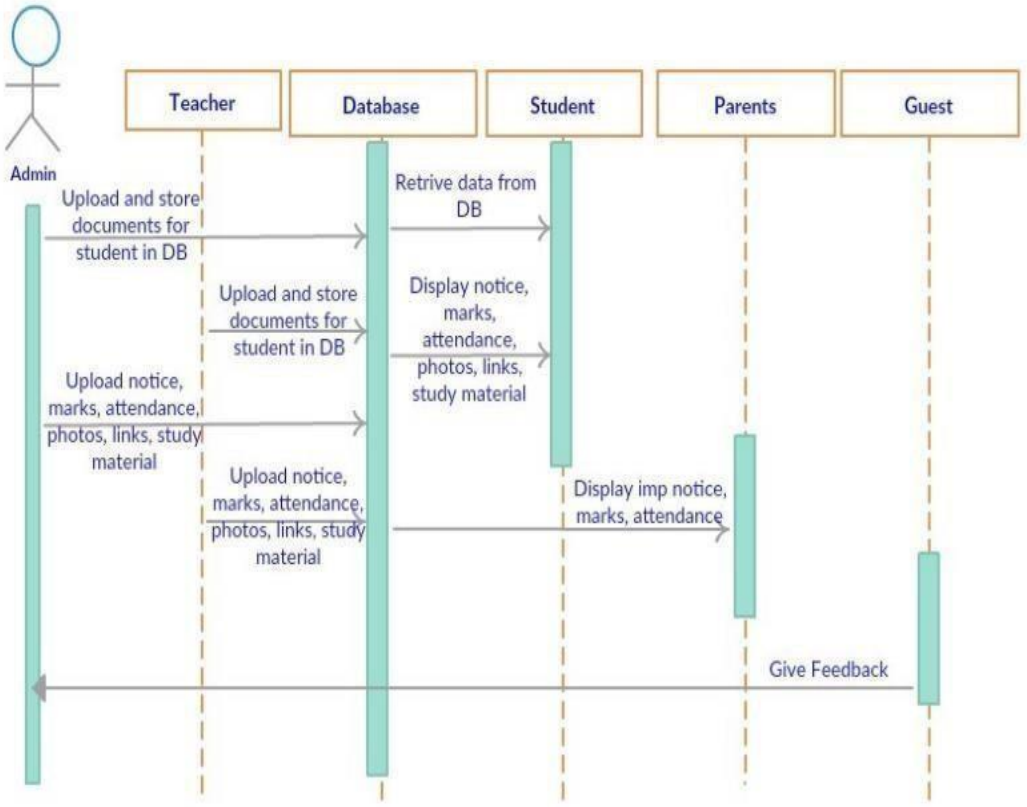

Fig 5.8 Sequence Diagram

A sequence diagram is a type of UML (Unified Modeling Language) diagram that represents the interactions between objects in a system over time. In the context of the chatbot system for college enquiry using a knowledgeable database, a sequence diagram can be used to represent the sequence of interactions between the user and the system when making a query. The sequence diagram for the chatbot system can include the user object, query object, response object, and the various algorithms and components of the system such as the Natural Language Processing Engine, Knowledgeable Database, Retrieval-based Algorithm, Rule- based Algorithm, Machine Learning Algorithm, Hybrid Approaches, and Feedback Mechanism.

5.8 COMPONENT DIAGRAM

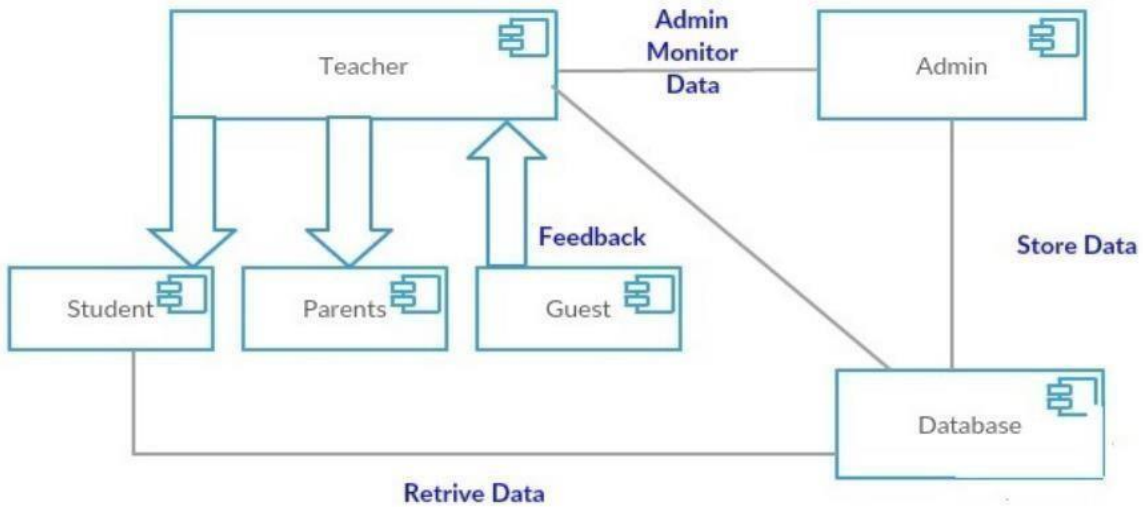

Fig 5.9 Component Diagram

A component diagram is a type of UML (Unified Modeling Language) diagram that represents the physical and logical components of a system and their relationships. In the context of the chatbot system for college enquiry using a knowledgeable database, a component diagram can be used to represent the various components of the system and their relationships. The component diagram for the chatbot system can include components such as User Interface, Natural Language Processing Engine, Knowledgeable Database, Retrieval-based Algorithm, Rule- based Algorithm, Machine Learning Algorithm, Hybrid Approaches, and Feedback Mechanism. Each component can have its own set of interfaces, ports, and dependencies that define its behavior and interactions with other components.

5.9 DEPLOYMENT DIAGRAM

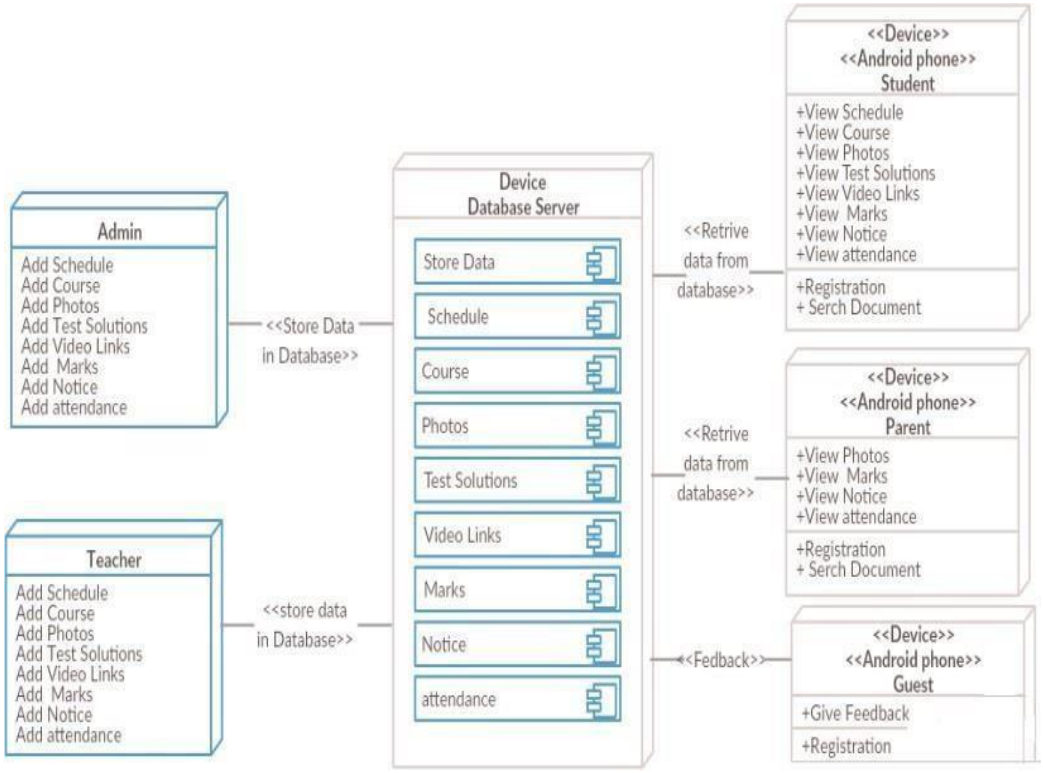

Fig 5.10 Deployment Diagram

A deployment diagram is a type of UML (Unified Modeling Language) diagram that represents the physical deployment of components and their relationships in a system. In the context of the chatbot system for college enquiry using a knowledgeable database, a deployment diagram can be used to represent the physical deployment of the various components of the system. The deployment diagram for the chatbot system can include nodes such as User Interface Node, Natural Language Processing Engine Node, Knowledgeable Database Node, and Feedback Mechanism Node. Each node can represent a physical machine or a logical grouping of machines that host the corresponding components.

5.10 COLLABORATION DIAGRAM

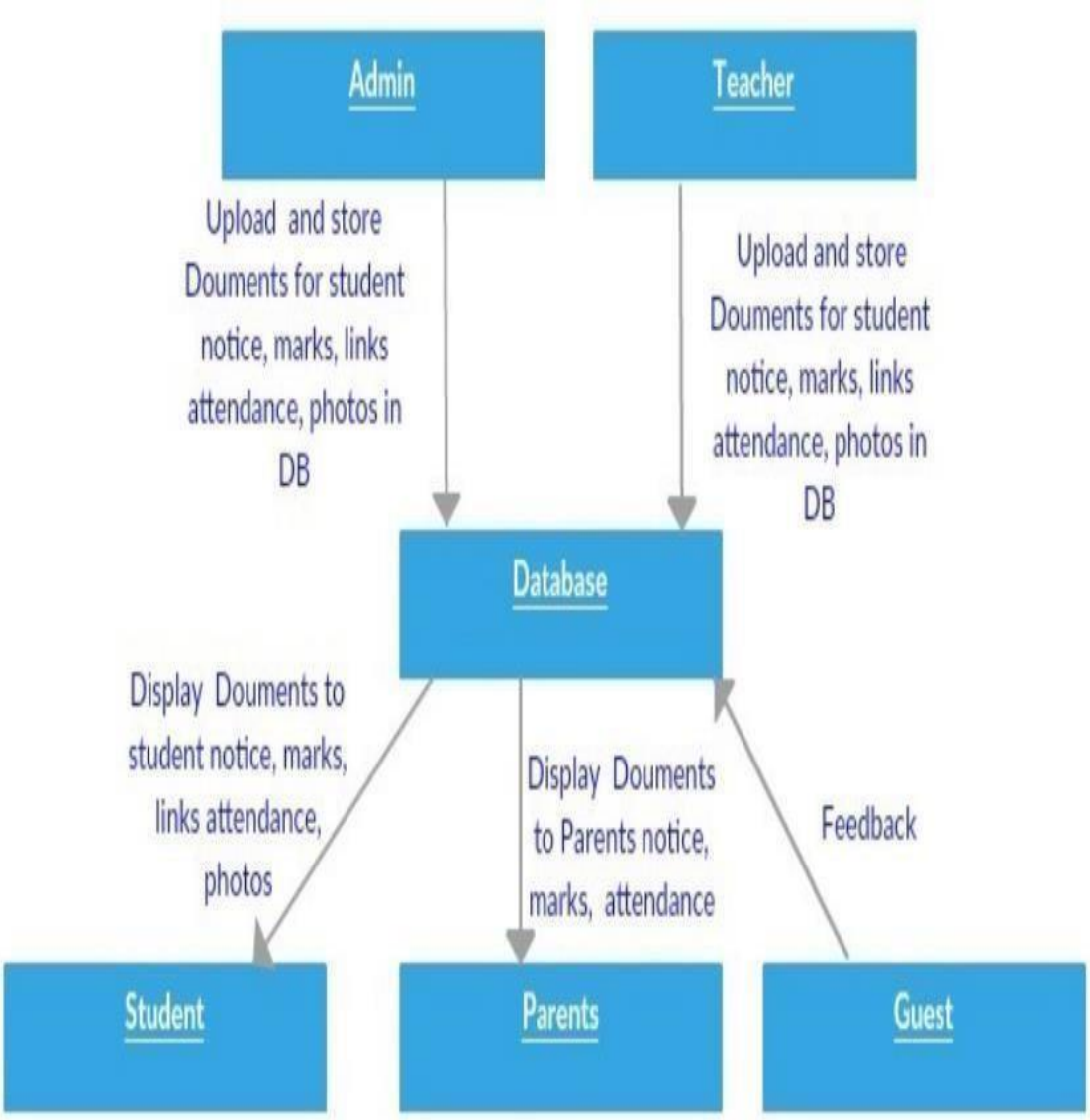

Fig 5.11 Collaboration Diagram

5.11 STATE CHART DIAGRAM

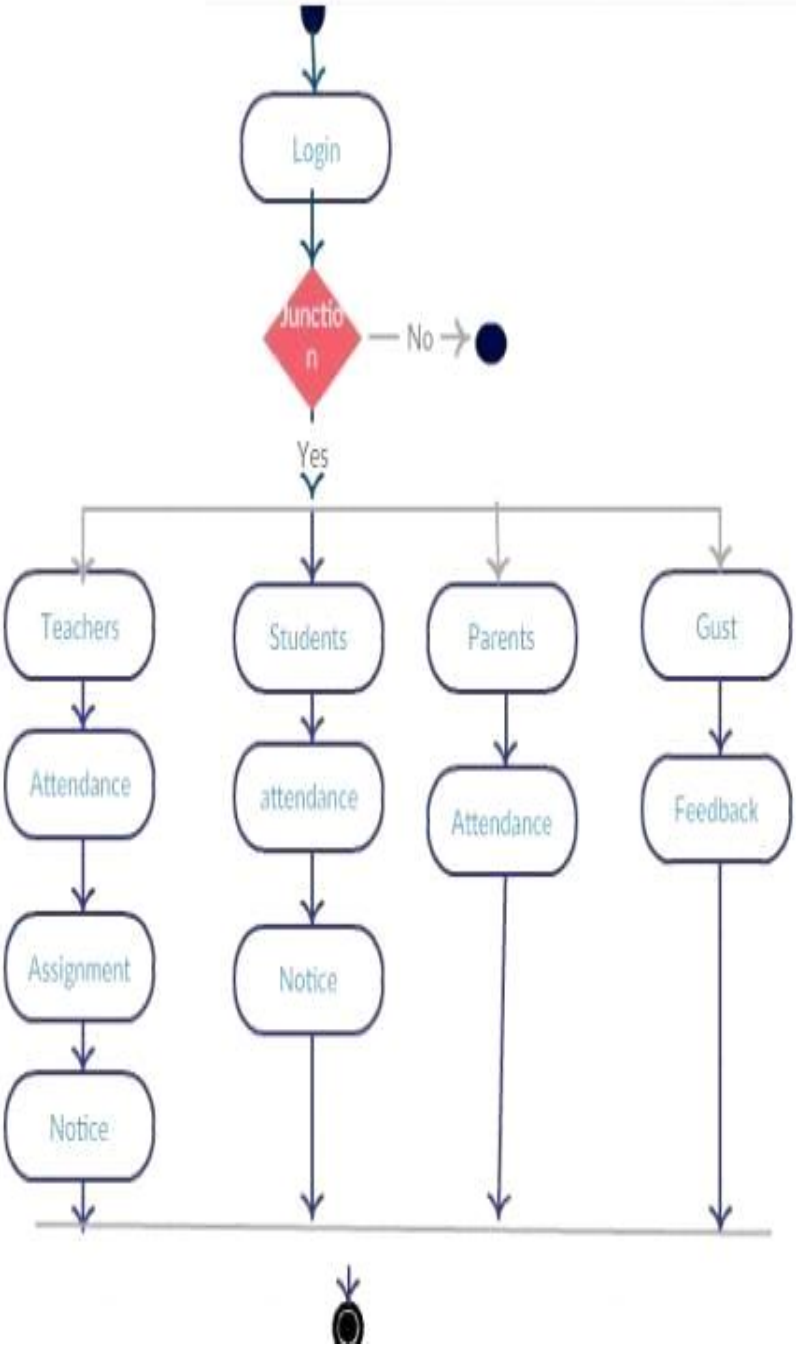

Fig 5.12 State Chart Diagram

CHAPTER-6

RESULTS AND DISCUSSION

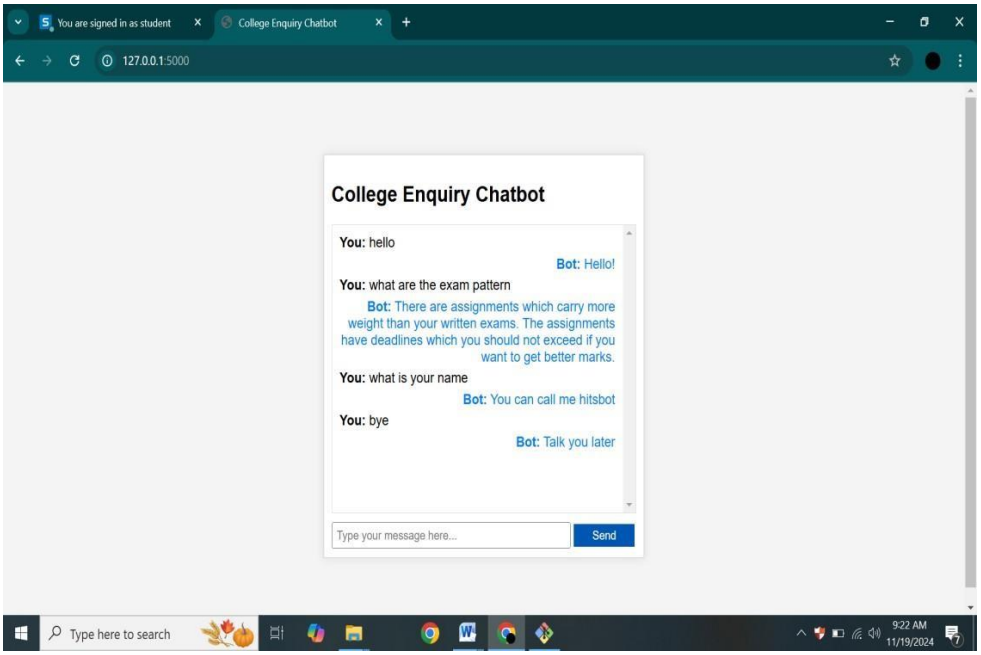

Fig.6.1. Execution (Output)

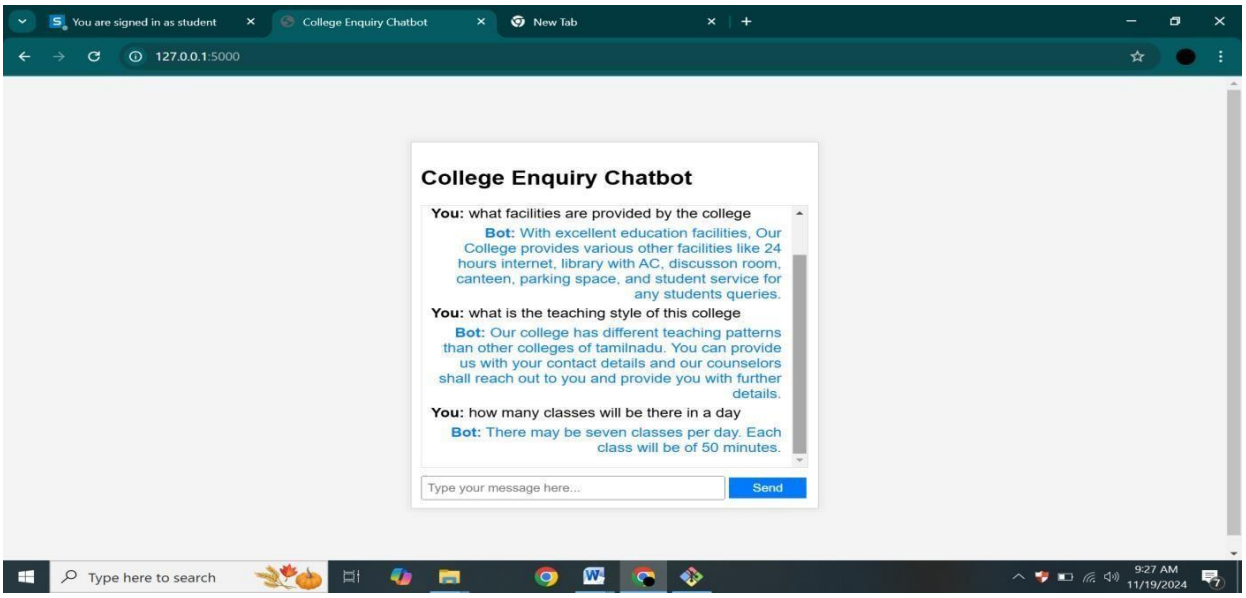

Fig.6.2. Execution (Output)

Our Chatbot provides information regarding to the college. It is the website. It is communicate to the client like guardians, understudy. By utilizing NLP human language changed into an information language. By utilizing AI to client give collegedata. This could be type-based (composed) discussion, even a non-verbal discussion. At the point when ChatBot innovation is incorporated with well known web administrations it very well may be used safely by a significantly bigger crowd. Chabot framework is carried out to meet scholarly necessities of the clients. Generating reaction from a Chabot is information Word Net is answerable for recovering the reactions and for this situation; it contains all rationales that is set off at whatever point the client setting is coordinated. At the point when a client starts asking questions in the Chabot Graphical User Interface (GUI). The question is looked in the information base..On the off chance that the reaction is found in the information base it is shown to the client else the framework tells the administrator about the missing reaction in the data set and gives a predefined reaction to the client.

Several studies have been conducted on college enquiry chatbots, and the results suggest that chatbots can significantly improve the efficiency and effectiveness of college enquiries.
Here are some brief results and discussions from these studies:

Improved efficiency: Chatbots can significantly reduce the workload on college administrators and provide faster, more efficient, and personalized services to students. For example, a study by Turel et al. (2021) found that a chatbot developed for student admissions reduced the average response time from 3 days to less than1 minute.

Higher user satisfaction: Several studies have found that students are generally satisfied with the performance of college enquiry chatbots. For example, a study byStieger et al. (2020) found that students rated the chatbot developed for their university highly on ease of use, usefulness, and overall satisfaction.

Accuracy and effectiveness: Chatbots have been shown to be effective in handling a wide range of college enquiries, including admission inquiries, course registration, financial aid, campus facilities, and career services. However, accuracyand effectiveness can vary depending on the quality of the chatbot's NLP and ML algorithms.

Challenges in chatbot development: Developing an effective college enquiry chatbot is not without challenges. Challenges include accurately identifying user intent, managing a large knowledge base, providing multilingual support, maintaining context across conversations, and ensuring a positive user experience.

Future research directions: Several research directions have been proposed for college enquiry chatbots, including improving NLP and ML algorithms, designing chatbots that can handle complex and multi-turn conversations, providing personalized recommendations and support, and developing chatbots that can handle emotional and mental health inquiries.

Overall, the results and discussions from the literature suggest that college enquiry chatbots , providing faster and personalized services to students. However, there is still much work to be done to improve the accuracy and effectiveness of chatbots and to address the challenges in chatbot development. Chatbots can gather data onuser queries, preferences, and behavior, which can be used to improve the chatbot's performance and inform college decision- making.

Chatbots can provide a more conversational and This can lead to increased user engagement and satisfaction. Chatbots can handle routine and repetitive enquiries, freeing up staff time to focus on more complex queries and tasks.Chatbots can be accessed anytime and anywhere through a range of devices, making it easier for students to get the information they need. Chatbots can be designed to provide personalized responses based on the user's profile, interests, and previous interactions with the chatbot.

Chatbots can handle multiple enquiries simultaneously, providing quick and efficient responses to users. However, there are also some challenges and limitations to the implementation of college enquiry chatbot.

CONCLUSION

7.1 CONCLUSION

Fastest-growing technology in history is artificial intelligence. utilizing a database that is both artificially intelligent and knowledgeable. We are able to transform virtual aid and pattern matching. This method is creating a chatbot based on the Android operating system with the help of a virtual assistant and an artificially intelligent database. A chatbot that can distinguish between human and machine speech andanswers to user enquiries is something we can make. Researchers must cooperateand decide on a common strategy in order to build a chatbot. In this study, we investigated the development of chatbots and their applications across several industries. Also, there are parallels with other chatbots. The knowledge base of the chatbot should generally be brief, approachable, and simple to understand. Even if some of the commercial solutions have just become accessible, there is still work tobe done in order to discover a standard method for building chatbots. In conclusion, a chatbot system for college enquiry using a knowledgeable database can provide a convenient and efficient way for students, faculty, and other stakeholders to access information about college programs, courses, and admission requirements. By utilizing natural language processing algorithms, a knowledgeable database, and a recommendation engine, the chatbot system can generate accurate and relevant responses to user queries in a timely manner. Save timing of students and teachers and also save extra manpower. Student can see all document related college like, notice, study material, question papers etc. on time to time and from any place. It is proper communication in between staff and students.

7.2 FUTURE WORK

As stated in the paper, the project has a broad reach in the current context. The proposal's majority of proposed features have been implemented. So, if I continue working on this project, I intend to create a database for the system where the admi may keep the extracted data. Further, future study will include a more in- depth examination of certain techniques, further research on other libraries, and new approaches to explore different methods.

Integration with other systems: Integrate with student information systems, CRM, or ERP systems for seamless data exchange.

Personalization: Use machine learning to offer personalized college recommendations based on users' interests, academic background, and career goals.

Multilingual support: Add support for multiple languages to cater to a broader audience.

Voice-based interface: Develop a voice-based interface to interact with the chatbot using voice commands.

APPENDIX

A. SOURCECODE

```
from flask import Flask, render_template, request, jsonifyimport random
import json import pickle
import numpy as npimport nltk
from nltk.stem import WordNetLemmatizer
from tensorflow.keras.models import load_model
app = Flask(__name__)
# Initialize lemmatizer
lemmatizer = WordNetLemmatizer()
# Load intents and model
intents=json.loads(open('intents.json').read())
word  = pickle.load(open('words.pkl', 'rb'))
 classes= pickle.load(open('classes.pkl', 'rb'))
model= load_model('chatbotmodel.h5')
# Function to clean up and tokenize sentence def clean_up_sentence(sentence):
sentence_words = nltk.word_tokenize(sentence)
sentence_words = [lemmatizer.lemmatize(word.lower()) for word insentence_words]
return sentence_words
# Function to create a bag of wordsdef bag_of_words(sentence):
sentence_words = clean_up_sentence(sentence)bag = [0] * len(words)
for w in sentence_words:
for i, word in enumerate(words):
if word == w:
```

```
bag[i] = 1

return np.array(bag)


# Function to predict class def predict_class(sentence):


bow = bag_of_words(sentence)

res = model.predict(np.array([bow]))[0]

ERROR_THRESHOLD = 0.25

results = [[i, r] for i, r in enumerate(res) if r > ERROR_THRESHOLD] results.sort(key=lambda x:

x[1], reverse=True)

return_list = [{'intent': classes[r[0]], 'probability': str(r[1])} for r in results]return return_list


# Function to get a response def get_response(intents_list, intents_json):if intents_list:

tag = intents_list[0]['intent'] for
```

```
i in intents_json['intents']: if i['tag'] == tag:
return random.choice(i['responses'])return "Sorry, I didn't understand that."

@app.route('/'

)def index():

return render_template('index.html')

@app.route('/get-response', methods=['POST'])def get_bot_response():
user_message = request.json.get('message')ints = predict_class(user_message) bot_response
= get_response(ints, intents) return jsonify({'response': bot_response})

if _____name__== '__main

_____':

app.run(debug=True) import json

import pickle

import numpy as npimport nltk nltk.download('punkt')

from nltk.stem import WordNetLemmatizer from

tensorflow.keras.models import Sequential from

tensorflow.keras.layers import Dense

Dropoutfrom tensorflow.keras.optimizers import SGD

# Initialize lemmatizer


lemmatizer = WordNetLemmatizer()
```

```
# Load data

with open('intents.json') as file:intents = json.load(file)

# Load pickle files

with open('words.pkl', 'rb') as file:words = pickle.load(file)
with open('classes.pkl', 'rb') as file:classes = pickle.load(file)


# Create training data def tokenize(sentence):
return [lemmatizer.lemmatize(word.lower()) for word innltk.word_tokenize(sentence)]


def bag_of_words(sentence): sentence_words = tokenize(sentence)bag = [0] * len(words)
for word in sentence_words:if word in words:
bag[words.index(word)]

= 1return np.array(bag)
X_train = []y_train
= []

for intent in intents['intents']: for pattern in
intent['patterns']: bag = bag_of_words(pattern) X_train.append(bag)
y_train.append(classes.index(intent['tag']))


X_train = np.array(X_train)y_
train
= np.array(y_train)

# Build and train the modelmodel = Sequential()

model.add(Dense(128, input_shape=(len(X_train[0]),), activation='relu'))model.add(Dropout(0.5))
model.add(Dense(len(classes), activation='softmax'))
```

```
model.compile(loss='sparse_categorical_crossentropy', optimizer=SGD(learning_rate=0.01),
metrics=['accuracy'])

model.fit(X_train, y_train, epochs=200, batch_size=5, verbose=1)



# Save the model model.save('chatbotmodel.h5')
print("Model trained and saved as 'chatbotmodel.h5'.")
```

```
{"intents": [
{"tag": "greetings",
"patterns": ["hello","hey","hi","good day","Greetings","what's up?","how is it going?"],
"responses": ["Hello!","Hey!","What can I do for you?"]
},

{"tag": "name",
"patterns": ["what is your name","name","what's your name","who are you","what should I call you"],
"responses": ["You can call me hitsbot","I'm hitsbot","I'm hitsbot your virtual assistant"]
},

{"tag": "courses",
"patterns": ["what courses are available", "how many courses are there in this college"],
"responses": ["For Bachelors in Business Administration we have been offering the followings:\n\n1.Bachelors in Technology we have been offering the followings:\n\n1.CSE\n\n2.MECH\n\n3.ECE\n\n4.EEE\n\n6.ARCH\n\n7.CIVIL"]
},

{"tag": "courseDuration",
"patterns": ["how long will be B.TECH course", "how long will it take to complete BBA or B.TECH course"],
"responses": ["Our college offers 4 year long B.TECH course."]
},

{"tag": "Location",
"patterns": ["location","where is it located","what is the location of the college"],
"responses": ["K.Ramakrishnan college of technology ,Samapuram ,trichy."]
```


,

```
{"tag": "studentRequirements",
"patterns": ["what are the student requirements for admission","entry requirements","admission requirements"],
"responses": ["Academic Level\nHSC +2 overall  55% ."]
},

{"tag": "classes",
"patterns": ["how many classes will be there in a day","how long are the classes?"],
"responses": ["There may be seven classes per day. Each class will be of 50 minutes."]
},

{"tag": "teachingStyle",
"patterns": ["what is the teaching style of this college?","Is the teaching pattern different from other college?","what is the teaching format?"],
"responses": ["Our college has different teaching patterns than other colleges of tamilnadu.\nYou can provide us with your contact details and our counselors shall reach out to you and provide you with further details."]
},

{"tag": "exams",
"patterns": ["what are the exams like?","What is the exam pattern"], "responses": ["There are assignments which carry more weight than your written
exams. The assignments have deadlines which you should not exceed if you want toget better marks."]
},

{"tag": "hours",
"patterns": ["what are your hours","when are you guys open","what your hours of operation"],
"responses": ["You can message us here at any hours. But our college premises will be open from 8:45 am to 4:30 pm only."]
},
```

A. SCREENSHOTS

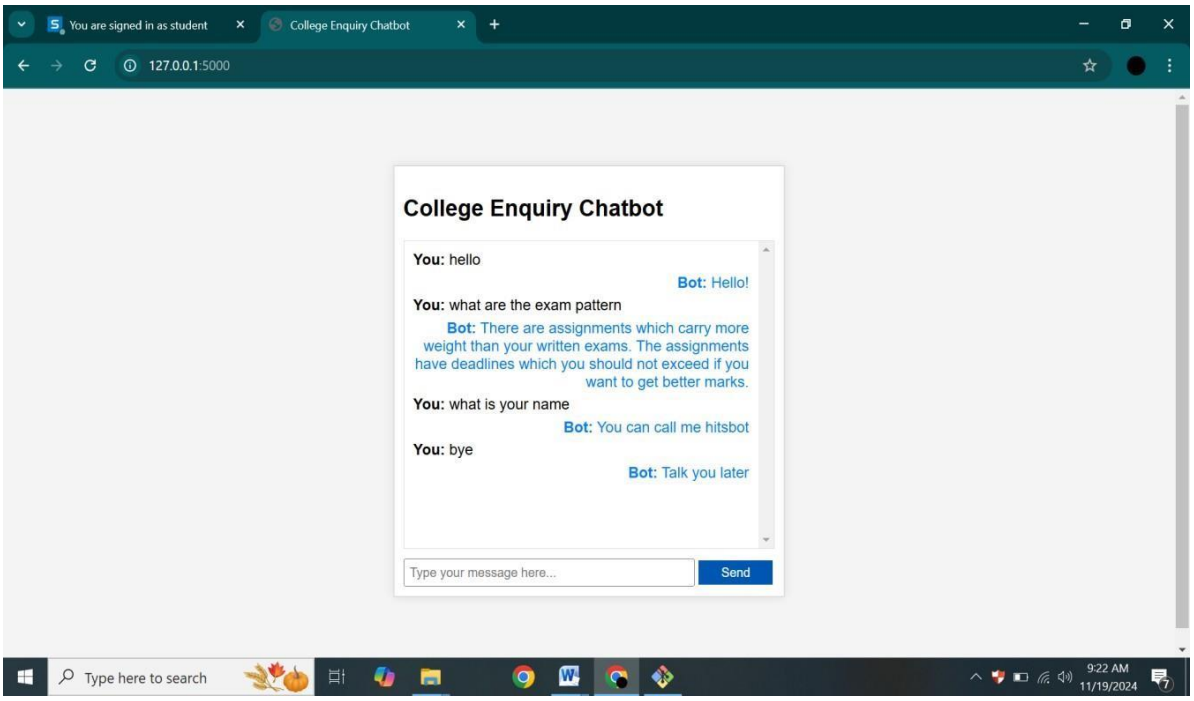

Fig.B.1. Execution (Output)

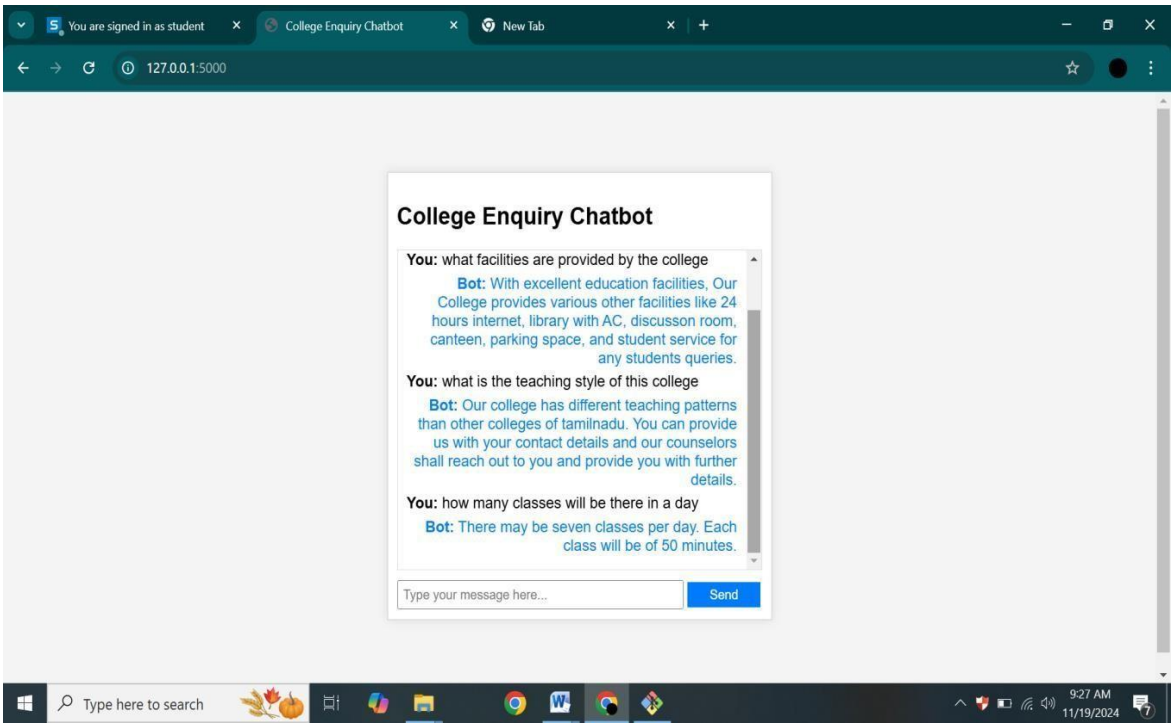

Fig.B.2. Execution (Output)

REFERENCES

[1] CHATBOT BASED ON AI Information technology professors Nikita Hatwarl, Ashwini Patil 2, and Diksha Gondane 3123 are from Nagpur/RTMNU in India. Volume 3, Issue 2 (March-April 2016), International Journal of Emerging Trends in Engineering and Basic Sciences (UEEBS), ISSN(Online)2349-6967)

[2] Chatbot Utilizing a Knowledge in Database at the 2016 7th InternationalConference on Intelligent Systems, Modelling and Simulation by Bayu Setiaji and Ferry Wahyu Wibowo.

[3] Chatbot, https://en.wikipedia.org/wiki/Chatbot.Schantz, Herbert F., The History of OCR, Recognition Technologies Users Association, ManchesterCenter, Vermont, 1982.

[4] P. Chenna, T. Rao, et al (2018). HFSS software is used to design a twin inverted L microstrip antenna for sustainable systems. International Journal of Computer-Aided Engineering Technology (IJCAET).

[5] J. Quintero, an IEEE student member, and R. Asprilla, an IEEE memberProceedings of the 2020 IEEE Thirty-Fifth Central American and Panama Convention: Towards an Efficient Voice-Based Chatbot (concapan xxxv).

[6] Rao, P.T., and Kumar, P.P. (2015,January). Twin microstrip patch antenna in the form of a staircase. 2015 saw the ICPC (International Conference on Pervasive Computing) (pp. 1-5). The Technical Writer's Handbook by IEEE
M. Young. University Science, Mill Valley, California, 1989.

[7] Rao, P. T., and Kumar, P. P. (2021). dual dual folded inverted-L antennawith a frequency-tunable circular polarisation and varactor-loaded split-ring resonator constructions. Global Journal of Communication Systems, e4715.

[8] Ly Pichponreay and Chi-Hwan Choi created the Chungbuk National University Smart Responding Chatbot based on OCR and Overgenerating Transformations and Rating.

[9] Trinatha Rao, P., and Praveen Kumar, P. C. (2020). Using HFSS, a double folded inverted-L antenna with a rectangular ground plane is designed to be reconfigurable. IETNetworks,9(5),229-234.

[10] Automated Question Answering System Utilizing Ontology and Semantic Role International Conference on Innovative Mechanisms for Industrial Applications (ICIMIA2017) by S. Jayalakshmi and Dr. Ananthi Sheshasaayee.

[11] Yuriy Dyachenko and Nayden Nenkov Articial Intelligence Technologies For Personnel Learning Management Systems 2019 IEEE 8th Internatignal Conference on Intelligent Systems Volodymyr DahlEast Ukrainian National University Severodonetsk, Ukrainev.

[12] "Development of Chatbot System for College Enquiry" by J. Rajitha and

S. Sathish Kumar. International Journal of Engineering Research & Technology,vol. 6, no. 3, March 2017.

[13] "Chatbot Based Enquiry System for College" by S. Sowmiya and S. Sathish Kumar. International Journal of Innovative Research in Science, Engineering and Technology, vol. 6, no. 8, August 2017.

[14] "Development of Chatbot for College Enquiry System using AIML" by R. Arun and A. Shanthi. International Journal of Computer Science and Mobile Computing, vol. 6, no. 9, September 2017.

[15] "Chatbot Based Enquiry System for College Admissions" by S. S. Gupta and R. K. Singh. International Journal of Computer Sciences and Engineering, vol. 6, no. 2, February 2018.

[16] "An AI-Based Chatbot System for College Enquiry" by N. Nidheesh et al.(2020). This paper describes the development of an AI-based chatbot system for college enquiry that utilizes a knowledgeable database and machine learningtechniques to provide accurate responses to user queries.