EDU-HELPER CHATBOT

A Project Report

Submitted to Goa University

In partial fulfillment of the Requirement

For the degree of

Bachelor in Computer Applications

BY

Mr. Vishal Naik

Mr. Devashish Parsekar

Mr. Gopal Harichandra parwar

Mr. Rahul Raju Pereira

Mr. Deepak Prasad

Mr. Rahulkumar

Ms. Sayali Vaman Redkar

Guided by

Ms.Milan Gaonkar

SPES Shri Gopal Gaonkar Memorial Goa Multi-Faculty College

Affiliated to Goa University

CERTIFICATE

This is to certify that the project on

" Edu-Helper Chatbot "

has been successfully completed and submitted by

Mr. Vishal Naik

Mr. Devashish Parsekar

Mr. Gopal Harichandra parwar

Mr. Rahul Raju Pereira

Mr. Deepak Prasad

Mr. Rahulkumar

Ms. Sayali Vaman Redkar

Ms. Milan Gaonkar

(Internal guide)

Ms. Nisha Sawant

(Project Coordinator)

Dr. Shaikh M. Parvaz Al-Usmani

(Principal)

External

Shruthi Nair

(Examiner)

SPES Shri Gopal Gaonkar Memorial Goa Multi-Faculty College

Affiliated to Goa University

DECLARATION BY CANDIDATES

We declare that this project report has been prepared by me/us and to the best of my/our knowledge, it has not previously formed the basis for the award of any diploma or degree by this or any other University.

Roll No	Name	Signature
9331	Vishal Naik	
9332	Devashish Parsekar	
9333	Gopal Parwar	
9334	Rahul Pereira	
9336	Deepak Prasad	
9337	Rahulkumar	
9338	Sayali Redkar	

SPES Shri Gopal Gaonkar Memorial Goa Multi-Faculty College

Affiliated to Goa University

CERTIFICATE BY THE PROJECT GUIDE

This is to certify that the project report is the record of the whole work done by the candidates themselves under my guidance during the period of study and that to the best of my knowledge; it has not previously formed the basis for the award of any diploma or degree by this on any other university.

Name of college: SPES Goa Multi-Faculty College

Programme: Bachelor in Computer Applications (B.C.A.)

Academic Year: 2022-2023

Ms. Milan Gaonkar

Project Guide

ACKNOWLEDGEMENT

We would like to thank all the people who have helped us during the project work and without their help the project would not have been a success.We wish to express our sincere gratitude to all those individuals who led a helping hand in all the difficulties that we faced. First and foremost we would like to thank Ms. Milan Gaonkar, our project guide who has provided us valuable guidance and assistance.We thank her for all the support, suggestion, guidance and ideas that she shared with us.

We are also thankful to our course coordinator Ms.Nisha Sawant for her valuable support. Special thanks to our Principal Dr. Shaikh Mohammad Parvez Al-Usmani, whose help was very crucial at all, the stages.

We would also like to take the opportunity to thank all our faculty members who have consistently co-operated with us and always supported and encouraged us.We would like to thank whole BCA Department for providing us with support and motivation and all the tools needed in the course of this project.

TABLE OF CONTENTS

Lists Of Figures

CHAPTER:1

INTRODUCTION

Technology plays a massive role in the industry and daily chores. It serves a variety of purposes and is applied in a different way in different parts of the world. Recently, the public has been fantasized by Artificial Intelligence and Machine learning.

Artificial intelligence (AI) refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. The term may also be applied to any machine that exhibits traits associated with a human mind such as learning and problem-solving.. Artificial Intelligence simulates the cognitive abilities of a human. To be more precise and closely related to humans, the AI Chatbots are now replacing human responses with this software.

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

The term chatbot consists of two other terms - chat and bot. The meaning can be better understood by examining the two components separately.A Chatbot is a computerized program that acts like a colloquist between the human and the bot, a virtual assistant that has become exceptionally popular in recent years mainly due to dramatic improvements in the areas like artificial intelligence, machine learning and other underlying technologies such as neural networks and natural language processing.

The chatbots effectively communicate with any human being using interactive queries. Recently, there's been a massive increase in many cloud-based chatting bot services which have been made available for the development and improvement of the chatbot sector such as IBM Watson, Cleverbot, ELIZA chatbot and many others. These

conversational agents have become more responsive and the art of conversation between humans and robots over the past few years have improved drastically.

1.1 History

Alan Turing in 1950 proposed the Turing Test ("Can machines think?"), and it was at thattime that the idea of a chatbot was popularized . The first known chatbot was Eliza, developed in 1966, whose purpose was to act as a psychotherapist returning the user utterances in a question form . It used simple pattern matching and a template-based response mechanism. Its conversational ability was not good, but it was enough to confuse people at a time when they were not used to interacting with computers and give them the impetus to start developing other chatbots . An improvement over ELIZA was a chatbot with a personality named PARRY developed in 1972 . In 1995, the chatbot ALICE was developed which won the Loebner Prize, an annual Turing Test,in years 2000, 2001, and 2004. It was the first computer to gain the rank of the "most human computer" . ALICE relies on a simple pattern-matching algorithm with the underlying intelligence based on the Artificial Intelligence Markup Language (AIML), which makes it possible for developers to define the building blocks of the chatbot knowledge . Chatbots, like Smarter Child in 2001, were developed and became available through messenger applications. The next step was the creation of virtual personal assistants like Apple Siri, Microsoft Cortana , Amazon Alexa ,Google Assistant and IBM Watson .There was a rapid growth of interestin chatbots especially after the year 2016. Many chatbots were developed for industrial solutions while there is a wide range of less famous chatbots relevant to research and their applications.

1.2 Types of chatbots

- Menu/button-based chatbots

Menu/button-based chatbots are the most basic type of chatbots currently implemented in the market today. In most cases, these chatbots are glorified decision tree hierarchies presented to the user in the form of buttons. Similar to the automated phone menus we all interact with on almost a daily basis, these chatbots require the user to make several selections to dig deeper towards the ultimate answer. While these chatbots are sufficient for answering FAQs that makeup 80% of support queries; they fall short in more advanced scenarios in which there are too many variables or too much knowledge at play to predict how users should get to specific answers with confidence. It's also worth noting that menu/button-based chatbots are the slowest in terms of getting the user to their desired value.

- Linguistic Based (Rule-Based Chatbots)

If you can predict the types of questions your customers may ask, a linguistic chatbot might be the solution for you. Linguistic or rules-based chatbots create conversational automation flows using if/then logic. First, you have to define the language conditions of your chatbots. Conditions can be created to assess the words, the order of the words, synonyms, and more. If the incoming query matches the conditions defined by your chatbot, your customers can receive the appropriate help in no time. However, it's your job to ensure that each permutation and combination of each question is defined, otherwise, the chatbot will not understand your customer's input. This is why a linguistic model, while incredibly common, can be slow to develop. These chatbots demand rigidity and specificity.

- Keyword recognition-based chatbots

Unlike menu-based chatbots, keyword recognition-based chatbots can listen to what users type and respond appropriately. These chatbots utilize customizable keywords and an AI application - Natural Language Processing (NLP) to determine how to serve an appropriate response to the user.These types of chatbots fall short when they have to answer a lot of similar questions. The NLP chatbots will start to slip when there are keyword redundancies between several related questions.It is quite popular to see chatbots that are a hybrid of keyword recognition-based and menu/button-based. These chatbots provide users with the choice to try to ask their questions directly or use the chatbot's menu buttons if the keyword recognition functionality is yielding poor results or the user requires some guidance to find their answer.

- Machine Learning chatbots

Ever wondered what is a contextual chatbot? A contextual chatbot is far more advanced than the three bots discussed previously. These types of chatbots utilize Machine Learning(ML) and Artificial Intelligence (AI) to remember conversations with specific users to learn and grow over time. Unlike keyword recognition-based bots, chatbots that have contextual awareness are smart enough to self-improve based on what users are asking for and how they are asking it. For example,a contextual chatbot that allows users to order food; the chatbot will store the data from each conversation and learn what the user likes to order. The result is that eventually when a user chats with this chatbot, it will remember their most common order, their delivery address, and their payment information and merely ask if they'd like to repeat this order. Instead of having to respond to several questions the user just has to answer with 'Yes' and the food is ready! While this food ordering example is elementary, it is easy to see just how powerful conversation context can be when harnessed with AI and ML. The ultimate goal of any

chatbot should be to provide an improved user experience over the alternative of the status quo. Leveraging conversation context is one of the best ways to shorten processes like these via a chatbot.

- The hybrid model

Businesses love the sophistication of AI-chatbots, but don't always have the talents or the large volumes of data to support them. So, they opt for the hybrid model. The hybrid chatbot model offers the best of both worlds- the simplicity of the rules-based chatbots, with the complexity of the AI-bots.

- Voice bots

To make conversational interfaces even more vernacular, businesses are now beginning to use voice-based chatbots or voice bots. Voice bots have been on the rise for the last couple of years, with virtual assistants like Apple's Siri, to Amazon's Alexa, and why? Because of the convenience they bring. It's much easier for a customer to speak rather than type. A voice-activated chatbot brings frictionless experiences directly to the end customer.

1.3 Technology behind chatbots

The application programming interface, in short — API, makes it possible for your chatbot to interact between your app and users through text messages. The intelligence of your chatbot will be defined by the way you apply machine learning technology while implementing the bot itself. If you want to go deeper into the chatbot development question it is not enough to know only that. For example, you would be excited to find out that there are actually different generations of chatbot technologies allocated already. This will help you to see a wider picture of how these bots can be ranked by quality.

Some of them are:

- **Rules-based**: a user asks a particular question, a chatbot replies with a strict answer.
- **Under supervised AI**: lots of labeled data created to imitate particular situations and behavior of users so the chatbot could act accordingly.
- **Under adaptive AI**: requires that chatbot will learn from unlabeled data in addition to the possibilities of the two previous generations.

While chatbots from the first generation can perform simpler tasks and be built by ordinary programming languages. For the second and third generations, NLP steps in. The more complex AI techniques are used the more difference between the latter two generations.

1.4 Machine learning

As already implied, machine learning makes chatbots smarter and more human-like in their responses.With these innovative technologies, chatbots can recognize human texts based on the previous knowledge base and gather new layers of natural language. This is how they learn from users.

In general, chatbots become smarter with:

- **Semantic parsing** which is then converted in the understandable for a machine form
- **Automated planning** that reaches the set goal by a predefined set of actions
- **Natural language generation** that makes it possible for chatbots to respond in a human manner

1.5 Why chatbots are gaining Importance?

A chatbot is often described as one of the most advanced and promising expressions of interaction between humans and machines. These digital assistants streamline interactions between people and services, enhancing customer experience. Chatbots have a lot to offer to businesses. Especially if you determine its functionality based on the needs of your company. Below is the list of general benefits of chatbots.

- **Automate customer service**

Chatbots can easily answer all simple questions your customer tend to ask from time to time.

- **Save human resources**

Chatbots are not getting tired answering simple inquiries like humans do. Instead, human resources will be of use in more complex tasks.

- **Improve company image**

As chatbots are available 24/7 potential customers can get the desired information anytime they decide to make a purchase which creates a better company image among them.

- **Save costs**

Chatbots development costs less than app development. Not to mention, that you will save money by replacing the human factor for simple inquiries from customers.

- **Stand out among competitors**

Chabot tend to be adopted by most businesses in the near future, so being among the first ones will give you leadership in your sphere and build tech-savvy customers loyalty.

- **Speed up processes**

Chatbots are not limited by the number of conversations they can lead unlike humans do. It makes them more efficient and cost-effective.

- **Choose time- and cost-efficient**

With chatbots, you can turn more visitors into paying customers faster and easier comparing to human only support.

1.6 Chatbot for Education Industry-The Future

Artificially intelligent chatbots do not only facilitate student's learning process by making it more engaging, short and snappy and interesting but also assist teachers by easing out their teaching processes.Not only this, but chatbots can also take the workload off of the administrative staff as well.Simply, AI will result in massive growth for the education industry, benefit the overall teacher-student interaction and improve the classroom environment; apart from enhancing learning, it will make students more tech-savvy and ready to put their foot into the real world.

1.6 Applications for chatbots in Education

It is increasingly common for students at all levels to use some kind of messaging service to communicate with each other and, occasionally, with their teachers.These are standard features in platforms such as Google Classrooms, and other class management systems,

and allow an exchange that aims, fundamentally, to ask questions and obtain answers that help the learning process outside the classroom.

In the same way, more and more MOOCs and other online courses are incorporating access to forums and communication systems that allow consulting and discussing issues with teachers and other colleagues.And according to different studies, the possibility of carrying out critical discussions about the content being studied allows them to build a better understanding, favoring the learning process.

Using chatbots, this process could be replicated on a large scale, generating channels where students could discuss any topic with an "expert", ask questions, and reach conclusions that would improve their understanding of different topics.

1.7 Some of the innovations that chatbots can provide in the field of virtual education:

It detects the emotional state of the students which, when identified by the chatbots, can modify the response with language adaptation or even incorporating a joke.

Provides personalized learning, adapting to the rhythm of the student, according to their needs and specific requirements. This provides a more direct orientation, when sending information or solving queries in relation to a course.

It allows the teacher to reduce time invested in organization and execution of tasks since chatbots provide immediate answers, previously predesigned, to frequent questions of the students.

This time saved can be invested in research or projects pending for the course, as well as in supervision and motivation of the group.

Store and analyze data effectively when reviewing the evaluation and progress of students. As a consequence of the use of Artificial Intelligence, it helps students organize their time and assign tasks according to their objectives in an effective and accessible way.

Improves access to education. The automatic learning tool is oriented to formation and interaction, does not consider the resources, the language or the location of the student. It can be considered something like a "democratization of learning"

CHAPTER:2

ANALYSIS

2.1 Existing System

In our college exists only the manual way of asking the queries to the appropriate staffs which will be an inconvenient way for students since they could not clarify their doubts at the time they need.The students had to visit the college to enquire about details like courses ,fee structure ,admission process and other information's about the college ,which is a tiresome process as well as long process for both parents as well as students. Now a days there are many changes occurred in the Education system with help of advanced technology. Everything is happening over the internet without any trouble. In those days for enquiring about courses we have to visit the college, but as the days are passing away its completely changing. Collecting the course details, fee structure manually will be hectic procedure and it also needs a manpower. For reducing that manpower and avoid such difficulties and time consuming process many devices or systems were emerged day by day.

Disadvantages

I. Attendance related issues.

II. Time consuming process

III. May lead to communication gap between college and students

2.2 Proposed System

A Student bot project (EduHelper) is built that analyzes user's queries and understand user's message. This System is a web based chatbot which provides answer to the query of the student. Students just have to query through the bot which is used for chatting. The answers are appropriate what the user queries.This Student bot project is built using training machine learning algorithms that analyzes user's queries and understand user's message.It also uses Rule-based algorithm in which bot use a series of defined rules. These rules are the basis for the types of problems the chatbot is familiar with and can deliver solutions. The bot analyzes the question and then answers to the user. The bot answers to the query as if it is answered by the person. With the help of database available backend i.e trained data, based on that system answers the query asked by the students. The system replies using an effective Graphical user interface(GUI) which implies that as if a real person is talking to the user. The user can query about the college related activities through online with the help of this web application.This system helps the student to be updated about their attendance,courses,faculties and the other college related activities.

Advantages

I. User does not have to go personally to college office for the enquiry.

II. This application enables the students to be updated on attendance and other activities.

III. This application will save time of the students as well as teaching and non-teaching staffs.

2.3 SOFTWARE REQUIREMENTS

I. VISUAL STUDIO CODE

II. PYTHON 3.7.0

III. MYSQL-DATABASE

IV. HTML

V. BOOTSTRAP

VI. PACKAGES OF PYTHON

- CHATTERBOT==1.0.4
- FLASK
- CHATTERBOT-CORPUS

VII. MySQl Workbench

2.4 FEASIBILITY STUDY

Feasibility study and important outcome of the preliminary investigation is the determination that the system requested is feasible.A feasibility study is carried out to select the best system that meets the performance requirement.The feasibility study is both necessary and prudent prevail with the feasibility of the project at the earliest possible time.

Three key consideration involved in the feasibility analysis are:

I. Economical visibility

II. Technical feasibility

III. Operational feasibility

- **Economical feasibility**

Economical feasibility is the cost and logistical Outlook for this project. Thus, the developed system as well within the budget and this was achieved because most of the technology used as freely available. The economical study analysis the data to determine whether the cost ultimately profitable to the user.So it is not a difficult task to any user to analyse the data due to this it is economically feasible.

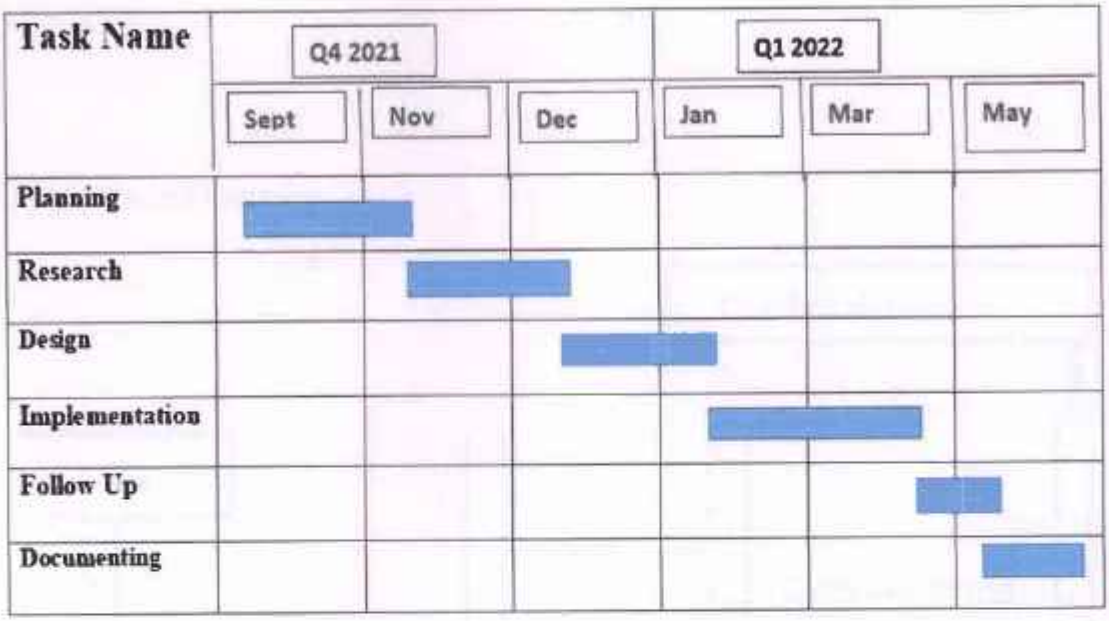
❖ **Technical feasibility**

Technical feasibility is one of the first studies that must be conducted after the project has been identified.Any system developed most not have high demand on the available technical resources.This led to high demand on the available technical resources.This lead to a high demand been placed on the client.This application has been developed with python where it provide the more general approach to data science.It is a general purpose language with a readable syntax.

❖ **Operational feasibility**

Assessing operational feasibility is the gain an understanding of whether the proposed system is solved the user problem. Or take advantage to the opportunity or not. Is important to understand how the new system will fetch into the current day-to-day operation of the organization.The operational feasibility studies are generally utilized to process, evaluation, implementation and resistance.The python also enable the developers to roll out program and get a prototype running. Making the development process much faster.

2.5 GANTT CHART



Task 1-Selecting and Planning Project

Task 2-Research about project

Task 3-Identifying scope and requirements of project

Task 4- Identifying Problem of existing System and Identifying the solutions

Task 5-System design analysis

Task 6-Drawing diagrams of system

Task 7-System Implementation

Task 8-System Testing

Task 9-Result Analysis

Task 10-Documentation

CHAPTER:3

DESIGN

3.1 Sequence Diagram

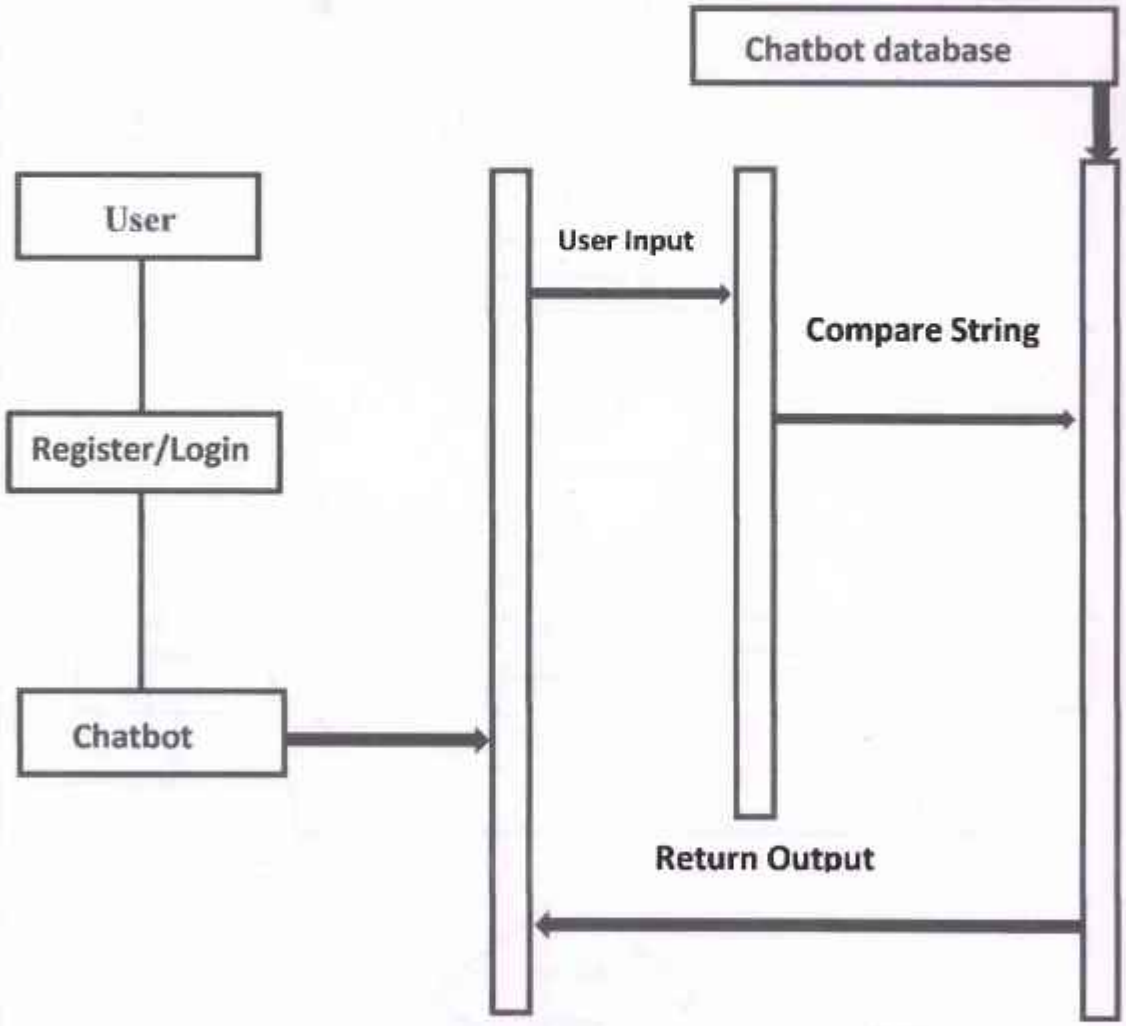


Fig. 1: Sequence Diagram Representing Design of the Chatbot Model

3.2 Process Flow Diagram

In this section, the basic steps regarding how we provide answers to the user queries will be shown in the following flow chart:

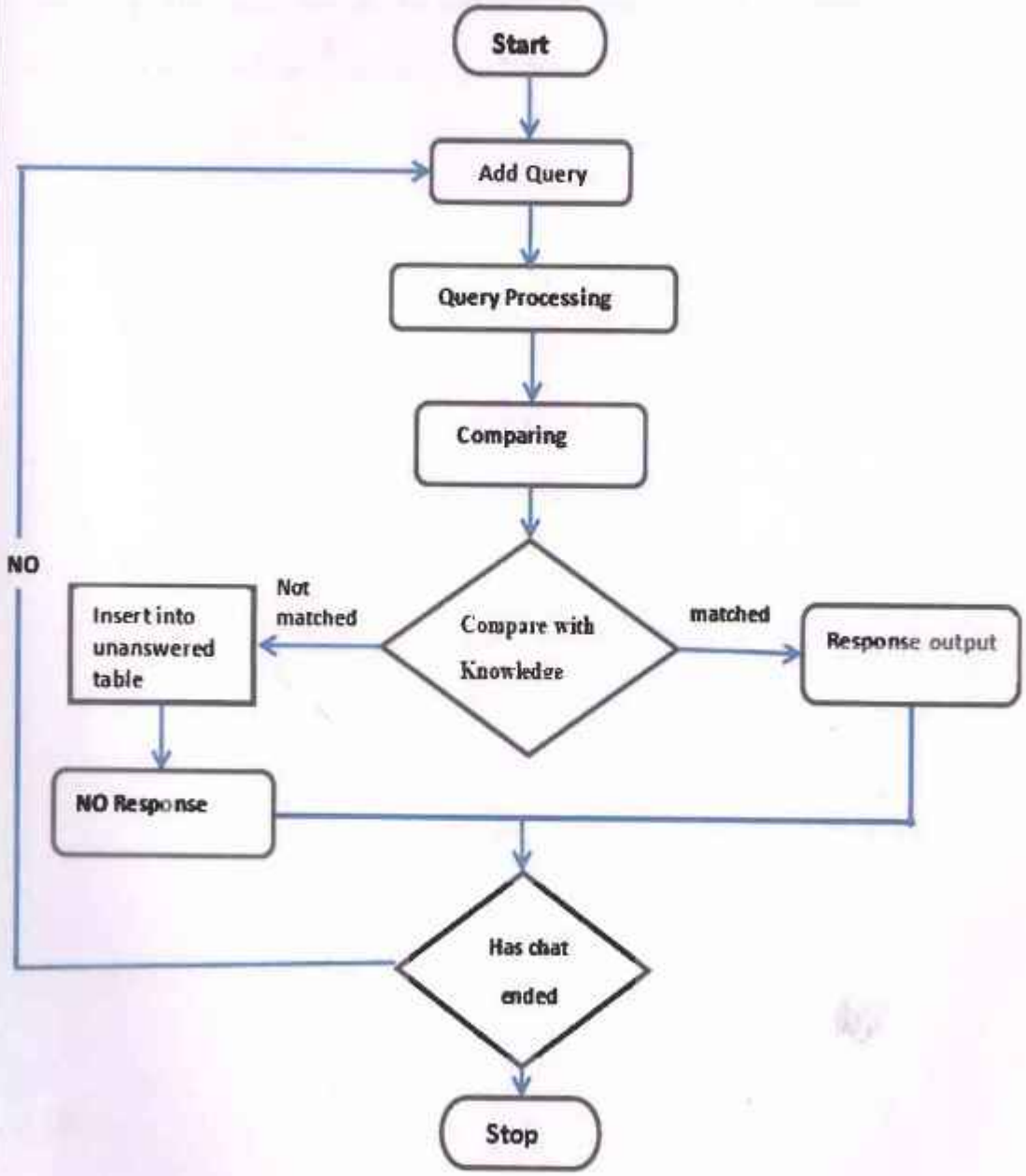


Fig-2: Flow Chart diagram for Edu-Helper Chatbot Model

The above Flow Chart describes the entire process of the system, if the query entered by the user is found in the database then the bot will give response.And if the user's query is not found in the database then bot will collect the details from the user and it will be stored in the database and one of the person from the college will contact the user personally. The questions will be stored in the database and the corresponding answers for those queries, will be updated by the Admin.

3.3 Use Case Diagram

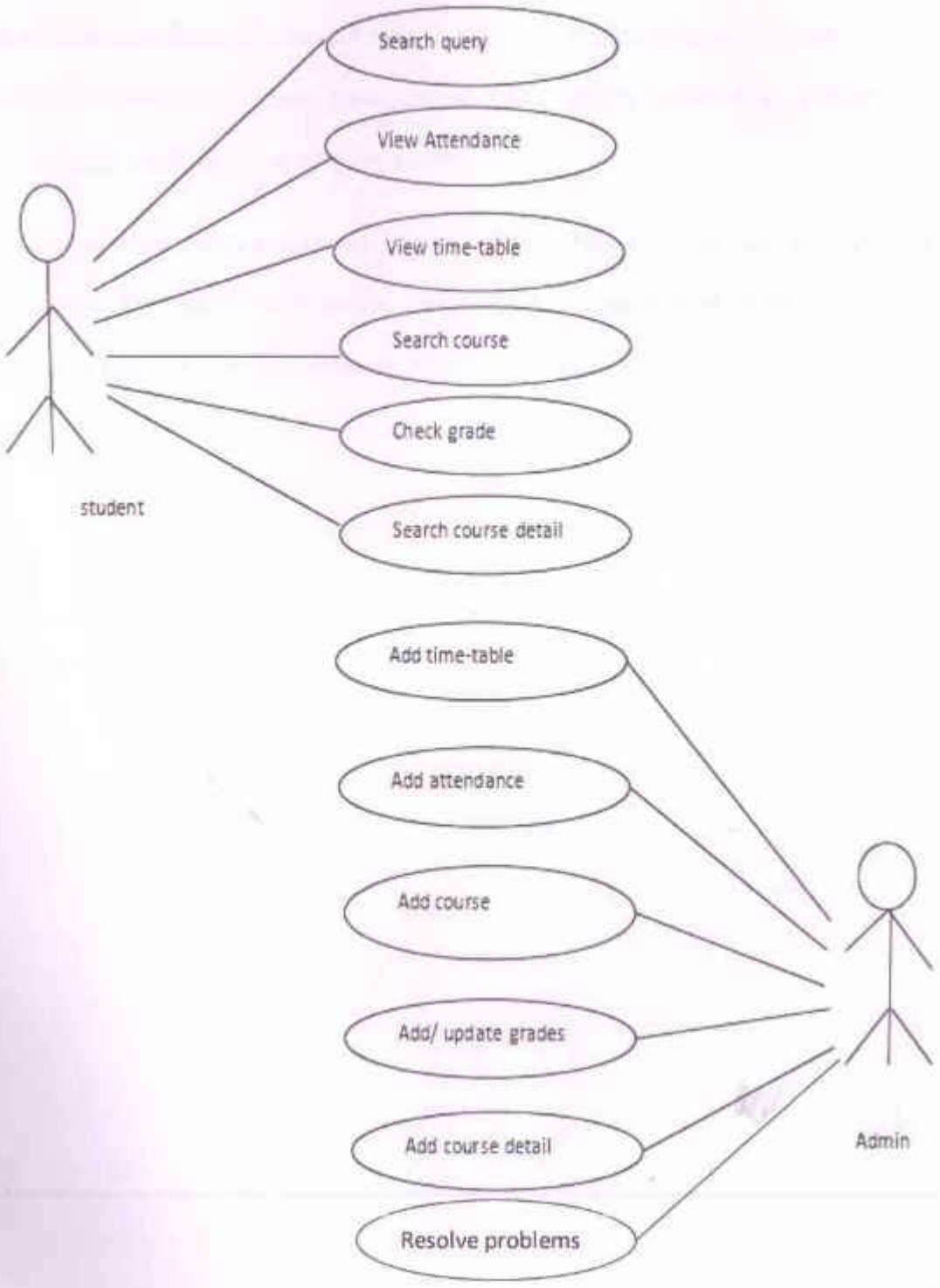


Fig-3: Use case Diagram of EduHelper Chatbot Model

In the above use case diagram there are two actors named students/users and admin. There are total of 12 use cases that represent the specific functionality of Chatbot each actor interacts with a particular use case. A students/users can interact with bot and check attendance, time table, course, grade, course detail, collage information. This actor can only perform this interaction with the system.

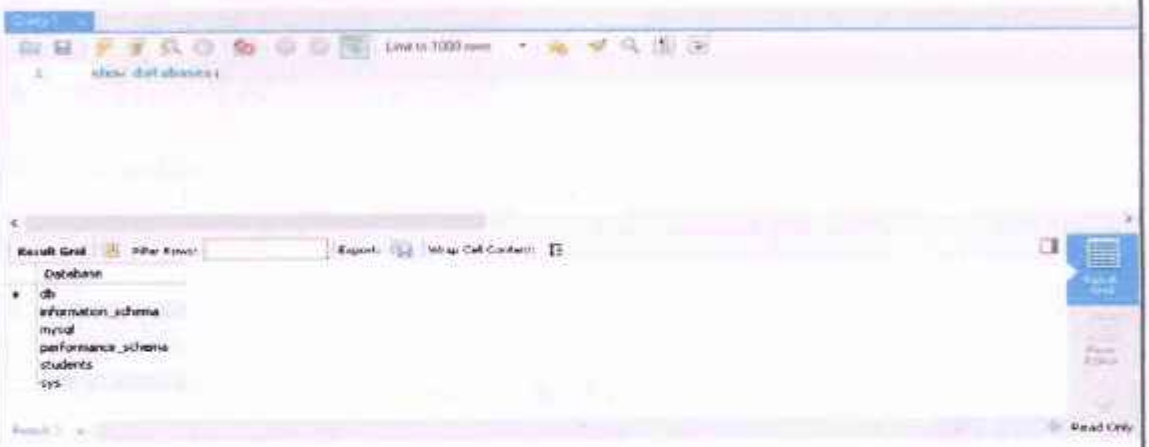
The second actor named admin can interact with functionalities like add time-table, Add attendance, Add course, Add/ update grades and Add course detail. Admin can add and update use cases. And he also resolves the students/users problems stored in unanswered database.

CHAPTER:4

Implementation

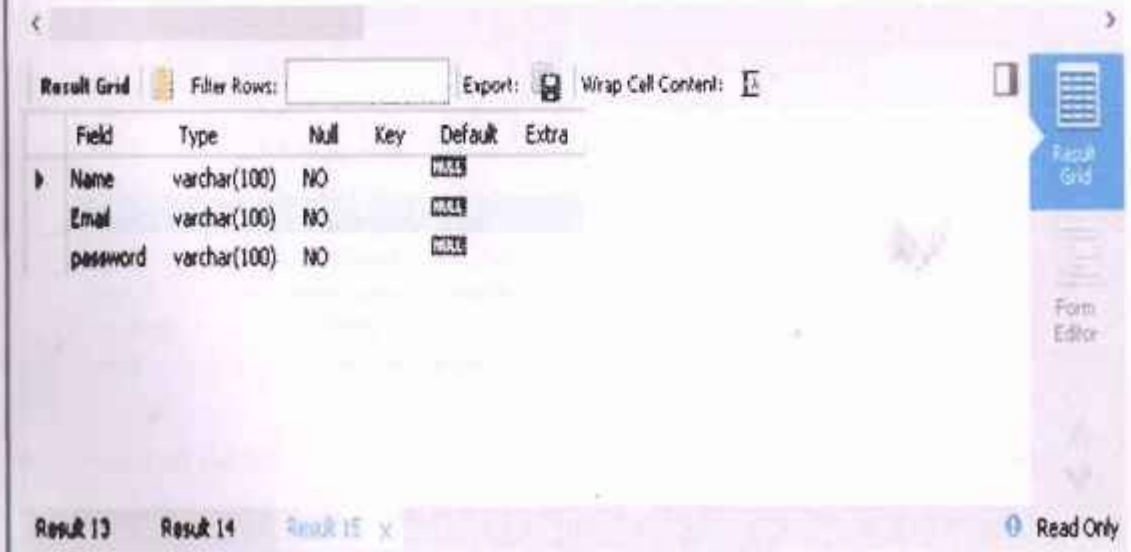
In this chapter the detail implementation of EduHelper-educational chatbot system is discussed.

4.1 Database Design



- Under student Database we have created sub database of Attendance,Login and Register

4.2 Registration database



Query:

- Show database ;
- Use to list all database on a MySQL server host.
- Use student ;
- Goes into student database
- Show tables ;
- To show the information in detail of registration
- Select * from register ;
- Is used to fetch the data from register database table which returns this data in the form of a result table.

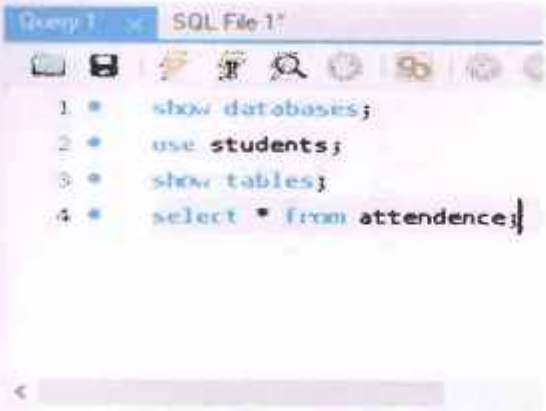
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

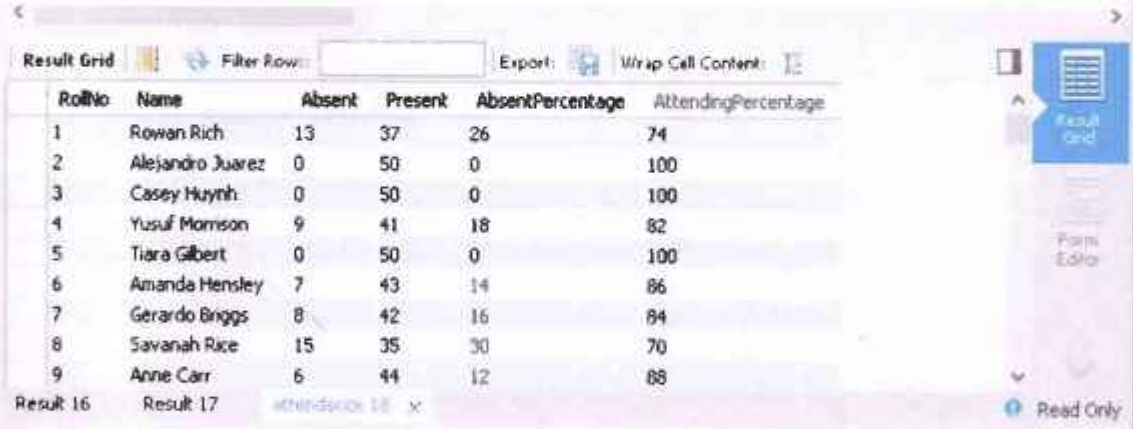
Name	Email	password
vishal	vishalnaik@gmail.com	welcome
devashish	devashishparsekar...	rangers
gopal	gopalparwar@gmai...	dragon
rahul	rahulpereira@gmail...	qwerty123
deepak	deepakprasad@gm...	hunter34
rahulkumar	rahulkumar@gmail....	hello123
sayal	sayalredkar@gmail...	ginger

➢ Based on the this, respective user who have registered .using same email and password they can do login .

4.3 Attendance Database

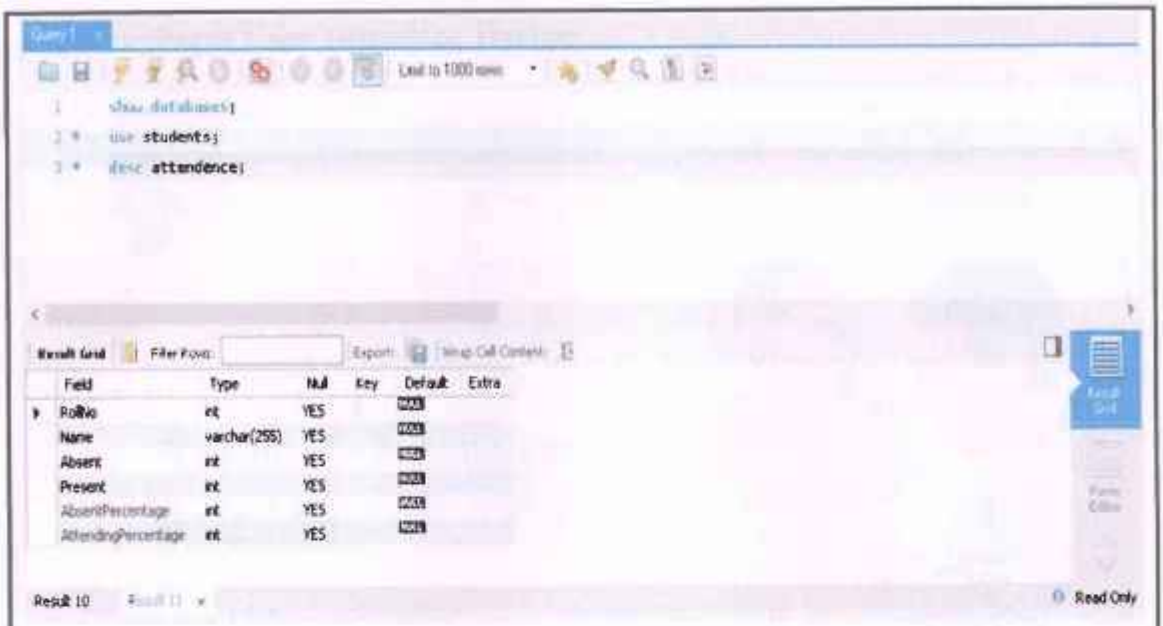
- Query to show Attendance Database





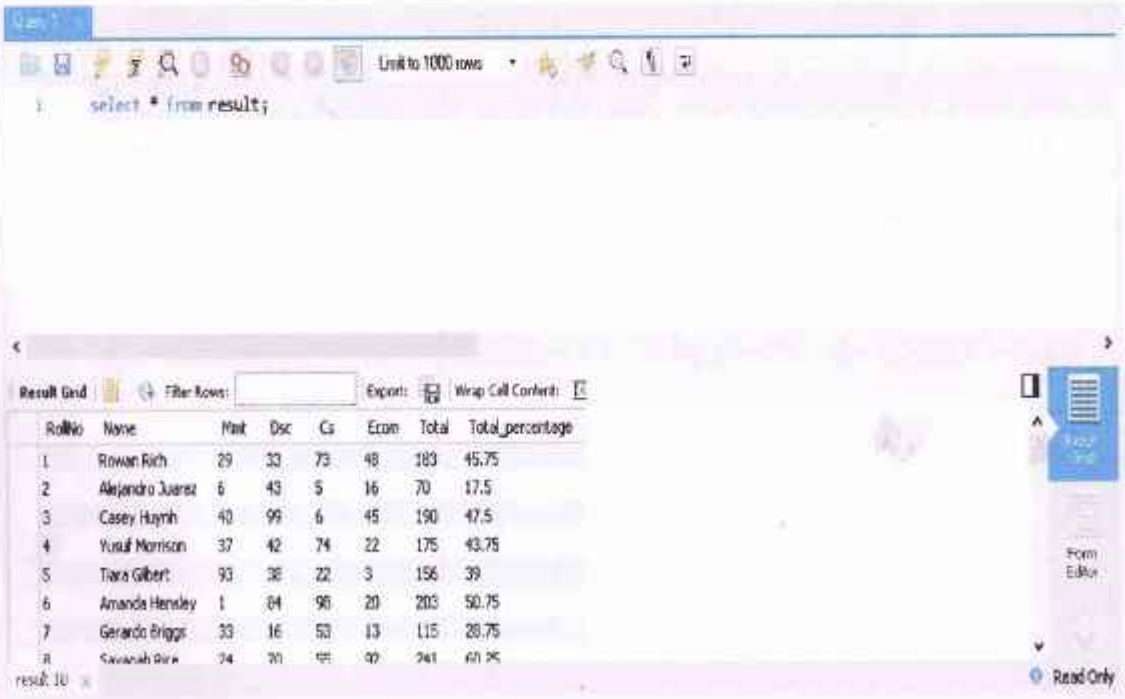
Select * from attendance ;

- Is used to fetch the data from attendance database table which returns this data in the form of a result table. It contain the Different column like Rollno,Name, Absent,Present and Attending percentage. Here you can see how many students absent and present and also percentage of absent and present.



➢ Shows the attendance table and describe the data type of attendance data in MySQL database. Eg (int, char, varchar)

4.4 Result Database Design

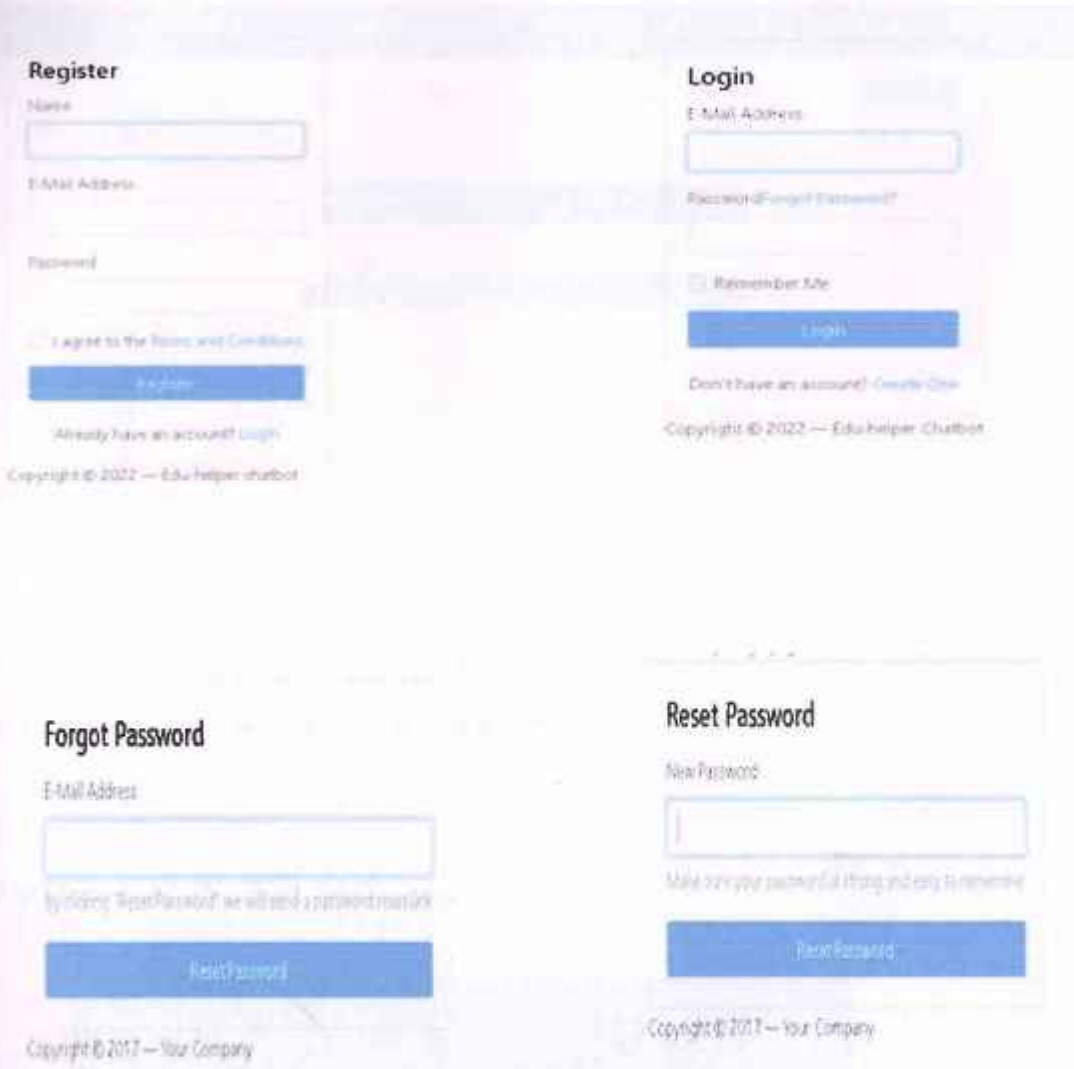


4.5 Graphical User Interface Design

This is home page of our web-based chatbot system .First users have to register and then do login to go to the chatbot.And then user can solve their query by asking question to chatbot. In this page user can contact us by Email or by Phone call to solve their query user can also follow us on Facebook, Twitter, Instagram and LinkedIn. User can also visit to various sections like Courses,About,Blogs,Faculty.

If you are new to this website then you can register yourself by entering your Name,email address and Password after filling all this details select I agree to all terms and Conditions and then click on Register Button. If you already have an account then Login by filling your email address and password.

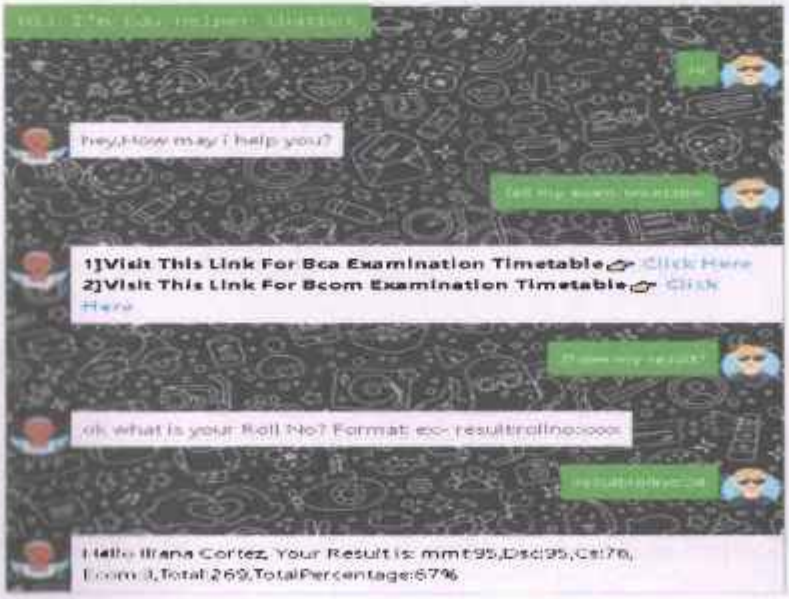
4.6 Interface for Registration and Login Page



4.7 Chatbot (GUI) Interface Design

After Login you will get Chatbot page form here you can solve your query by asking to Edu-helper chatbot. You can ask any thing like your attendance,subjects, courses,notice,timetable, etc.

Interface of chatbot while chatting



4.8 Instructional Coding

- **Installation**

The recommended method for installing ChatterBot is by using pip.

Installing from Python

pip install chatterbot = = 1.0.4

To install ChatterBot from Python using pip run the following command in your terminal.

- **Import the libraries**

```
4
5   from logging import WARNING, FileHandler,WARNING
6   from urllib import response
7   from flask import Flask, render_template, request
8   from chatterbot import ChatBot
9   from chatterbot.trainers import ChatterBotCorpusTrainer
10  from chatterbot.trainers import ListTrainer
11  from registeration import *
12  from train import *
13  from train1 import *
14  from getData import *
15  from registeration import *
```

we have imported various libraries to build to the chatbot.For example: Flask framework,chatterbot library,chatterbot-corpus.

- **create a new chatbot**

Create a new instance of the ChatBot class.

chatbot = ChatBot('Edu-helper Chatbot')

This line of code has created a new chat bot named Edu-helper chatbot.

specify before we run our program for the first time.

➢ Setting the storage adapter

chatbot=ChatBot("Edu-helperChatbot",storage_adapter="chatterbot.storage.SQLStorageAdapter", logic_adapter=["chatterbot.logic.BestMatch",'chatterbot.logic.MathematicalEvaluation','chatterbot.logic.TimeLogicAdapter'],preprocessors=['chatterbot.preprocessors.clean_whitespace'])

ChatterBot comes with built in adapter classes that allow it to connect to different types of databases. In this tutorial,we will be using the SQLStorageAdapter which allows the chat bot to connect to SQL databases. By default, this adapter will create a SQLite database.The database parameter is used to specify the path to the database that the chat bot will use. For this example wewill call the database **sqlite:///database.sqlite3**. this file will be created automatically if it doesn't already exist.

```
chatbot = ChatBot(
'Edu-helper Chatbot',
storage_adapter='chatterbot.storage.SQLStorageAdapter',
database_uri='sqlite:///database.sqlite3'
)
```

Note: The SQLStorageAdapter is ChatterBot's default adapter. If you do not specify an adapter in your constructor, the SQLStorageAdapter adapter will be used automatically.

➢ Specifying logic adapters

The logic_adapters parameter is a list of logic adapters. In ChatterBot, a logic adapter is a class that takes an input statement and returns a response to that statement.You can choose to use as many logic adapters as you would like. In this example we will use two

logic adapters. TheTimeLogicAdapter returns the current time when the input statement asks for it.

```
chatbot = ChatBot(
'Edu-helper Chatbot',
storage_adapter='chatterbot.storage.SQLStorageAdapter',
logic_adapters=[
'chatterbot.logic.TimeLogicAdapter'
],
database_uri='sqlite:///database.sqlite3'
)
```

➢ Training your chat bot

```
trainer = ChatterBotCorpusTrainer(chatbot)
trainer1 = ListTrainer(chatbot)
trainer1.train(data)
trainer1.train(data1)
trainer1.train(attendenceConvo)
trainer1.train(unrelateddata)
trainer1.train(data_of_human)
```

At this point our chat bot, Edu-helper Chatbot will learn to communicate as you talk to him. You can speed up this process by training him with examples of existing conversations.

- **This code combines all dataset in one variable.**

```
combined_data=data+data1+attendenceConvo+unrelateddata+data_of_human
```

- **checking of blank messages**

```
if userText == " ": return str("Please dont send blank messages")
```

This code will check blank spaces and if found then chatbot will give message to the user that"Please dont send blank messages".

- **This code accepts the complaint from students/user and store into complaint.txt file**

```
if userText[:11] == "complaint::":
    complaint_message=userText[11:]
    file1=open("complaint.txt","a")
    file1.write(complaint_message+"\n")
    file1.close()
    return str("Thank you your complaint is registered")
```

- **This code will show the attendance to the user**

```
If userText[:8] == "rollno::" and (log_of_chat[-2] == 'ok what is your Roll No:?Format:rollno::xxxx' or log_of_chat[-2] == 'ok what is your Roll No:?Format:rollno::xxxx'):
    RollNo=userText[8:]
    attendence=get_Attendence(RollNo)
    return str(attendence)
```

```
elif userText[:8] == "rollno::" and (log_of_chat[-2] != 'ok what is your Roll No: ? Format:rollno::xxxx' or log_of_chat[-2] != 'ok what is your Roll No:? Format:rollno::xxxx'):
        return str("Please first ask me to show your attendence")
```

In this code if user ask for attendance then bot will tell the user to put the roll no in the proper format .After putting the roll in the proper format chatbot will show the attendance of the respective user.

- **checking of user text in combined data**

```
if userText not in combined_data:
        if userText.capitalize() not in combined_data:
            #return str("If you giving new input than its good but first get the permission to add your new input")
            user_message=userText
            file2=open("user_did_not_get_answer.txt","a")
            file2.write(user_message+"\n")
            file2.close()
            return str("This type of input is not present in our database so i din't understand it")
```

In this code, chatbot check the user text in combined dataset and if user text is not found then it will capitalize the user text and again find into the combined dataset and if that text is not available again in combined dataset than it will return "If you giving new input than its good but first get the permission to add your new input"

- **This code will check bot response in not_to_reply list of data and if available in that data it will return :**

```
"If this input is correct than Sorry i am still learning")
if bot_response in not_to_reply:
    return str("If this input is correct than Sorry i am still learning")
log_of_chat.append(str(bot_response))
return str(bot_response)
```

- **Flask:**

from flask import Flask, render_template, request

we created a new flask app here

app = Flask(__name__)

we have imported the flask in our chatbot to communicate via website.

@app.route is a function of flask framework that can handles the user request

currently we are running the chatbot in loacalhost

127.0.0.1:5000

- **This statement will handle the homepage request**

for example: 127.0.0.1 is opened in browser than homepage will open

```
@app.route("/")
def index1():
return render_template("index.html")
```

- **The request for bot should be like 127.0.0.1/bot**

```
@app.route("/bot")
def bot():
    return render_template("index1.html")


@app.route("/index")
def index():
    return render_template("index.html")
```

- **If anyone is requested 127.0.0.1 than login page will open**

```
@app.route("/login")
def login():
    return render_template("login.html")


@app.route("/register")
def register():
    return render_template("register.html")


@app.route("/forgot")
def forgot():
    return render_template("forgot.html")
@app.route("/reset")
def reset():
    return render_template("reset.html")
```

- **Flask will collect data like Name,Email,Password and POST into register database.**

post_Data function will stored all the data in mysql after executing

```
@app.route('/register', methods =['GET', 'POST'])
def register_data():
  if request.method == 'POST' and 'name' in request.form and 'email' in request.form and 'password' in request.form:
      name=request.form['name']
      email=request.form['email']
      password=request.form['password']
      print(email)
      print(password)
      print(name)
      post_data=postData(name,email,password)
      print(post_data)
      if post_data:
        msg01="Registration Completed"
        return render_template('register.html')
```

when we creating a account on our website than after clicking on the register button all the data of name,email and password will posted on register route and this code will catch that post request than only it will get executed.Data will be stored in variable called name,email and password before that it will check if the post requested name, email and password is present in the submitted form.

- **Registration and Login: import mysql.connector**

when we are login in chatbot the email and password is checked after getting matched than only the chatbot will opened.

```
def check_login(Email=None,password=None):
    Try:
mydb=mysql.connector.connect(host="localhost",database="students",user="root",
password="root")
      mycursor=mydb.cursor()
      #mycursor.execute('delete from attendence where RollNo=1;')
      mycursor.execute("select * from register where Email='{}' and
password='{}';".format(Email,password))
      result=mycursor.fetchall()
      print(result)
   except mysql.connector.Error as e:
      print(e)
   finally:
      mydb.commit()
      if mydb.is_connected():
         mydb.close()
   return result
```

this function will post the data in mysql database called register with name email and password and it will get inserted

```
def postData(name=None,email=None,password=None):
   query=False
```

```
    if name == None or email == None or password == None:
        return None
    try:

mydb=mysql.connector.connect(host="localhost",database="students",user="root",
password="root")
        mycursor=mydb.cursor()
        mycursor.execute("INSERT INTO register (Name,Email,password) VALUES
('{}','{}','{}');".format(name,email,password))
    result=mycursor.fetchall()
        query=True
    except mysql.connector.Error as e:
        print(e)
    finally:
        mydb.commit()
        if mydb.is_connected():
            mydb.close()
    return query
```

After doing all integration.At final step execute **app.py** file in the terminal then in terminal user will get link to visit home page of our Chatbot Website .

CHAPTER:5

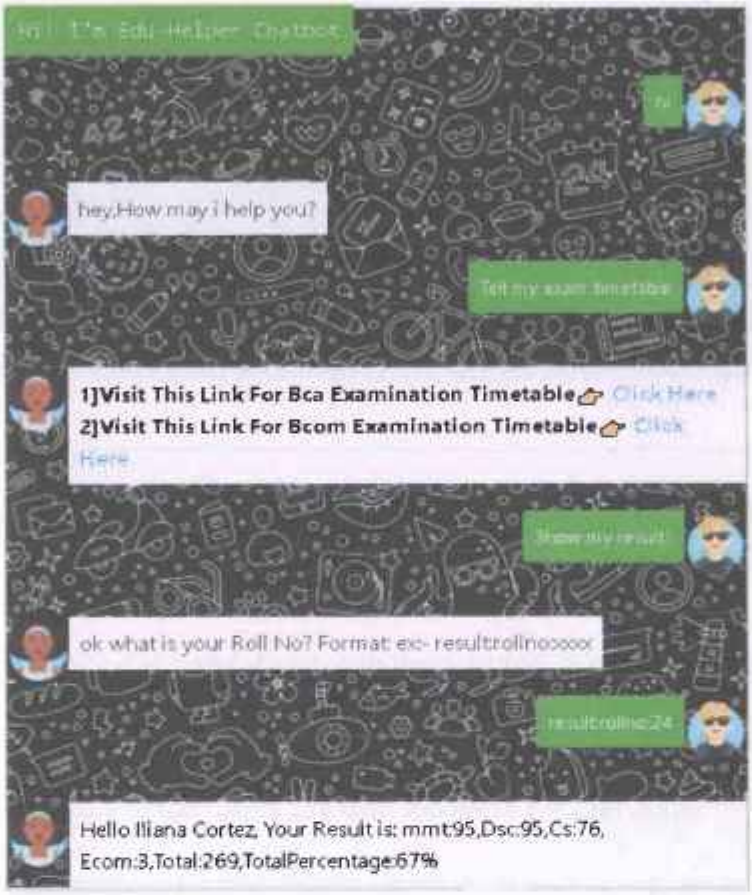
TESTING

5.1 Unit Testing

Unit testing involve the design of test cases that validates that the internal program logic is functioning properly,and that program input produces valid output. All decision branches and internal code flow should be validated.The testing of individual software unit of the application, The project is done after the completion of all and individual work before integration.Unit tests perform basic test at component level and test specific business process,applications,and or system configuration.Unit test ensure that each unique path of business process perform accurate to the documented specification and content clearly define input and expected results.

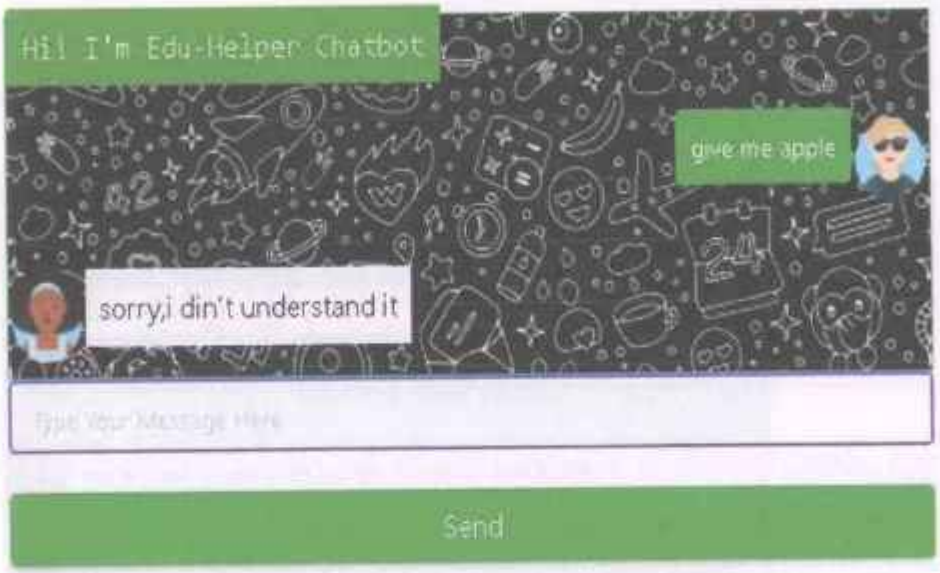
5.2 Validation Testing

NO	Input Details	Expected output	Actual output	Test case result
1	Ask a question	Provide answer	Provided	Pass
2	Enter a invalid sentence	Provide error	Provided	Pass
3	User satisfied by the answer	Yes	Yes	Pass
4	Ask a question with spelling mistakes	Spell checker identified error	Spell checker not identified error	Invalid
5	Interface-Display remains the same while chatting	Remains same	Remains same	Pass
6	Question not answered by bot stores in database	Yes	Yes	Pass

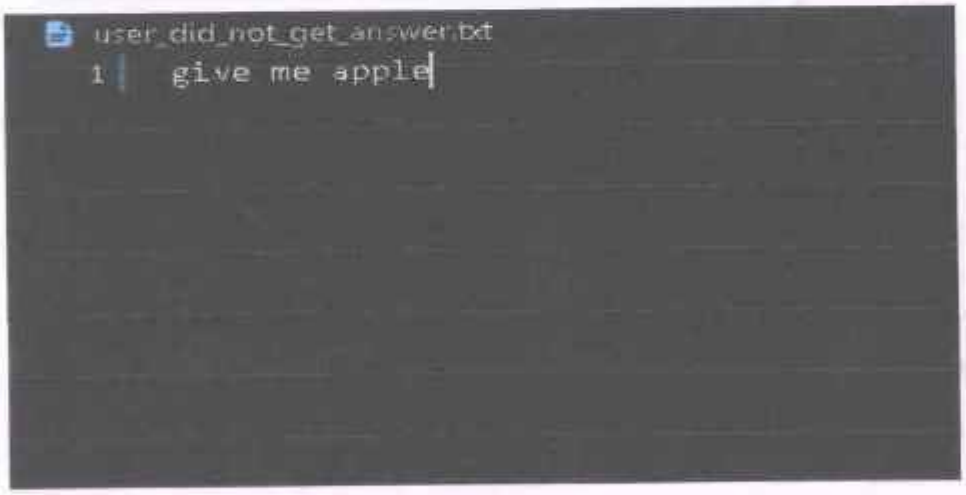


- If you ask question to the chatbot .It will provide you response.

- When user ask questions with spelling mistake then chatbot provide related Response.



- If user give unrelated input then bot will say "sorry, I didn't understand it" but that input will be stored in the user_did_not_get_answer.txt file and then admin can contact personally to the respective user and solve the user issues .



- **Interface remains the same doesn't show any changes while chatting with the systems.**

5.3 System Integration Testing

Integration test are designed to test integrated software components to determine if the user actually run as one program.Testing is event driven and is more concerned with the basic outcome of screen or fields.Integration test demonstrate that although the module were individually satisfactory, as shown by successfully unit testing,the combination of components is correct and consistent.Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

CHAPTER: 6

6.1 Limitations of the system

I. It require regular internet connection every time user cannot access the system without internet.

II. If many user use the system at the same time the response may be slow or even may crash due to server down.

III. Give systematic response for only pre-defined sets of questions.

6.2 Future Scope

To improve the current functionalities of college enquiry chatbot in the future. the scope of chatbot can be increased by :

I. Training the bot with veried data.

II. Speech recognition feature through which the student can ask their queries verbally and get the answer from the bot.

III. Can be used for global market

IV. Adding the capability for the chatbot to perform analytic based on the user sentiment based on which the bot can be re-train on the human emotions so that more empathy can be added to the chatbot.

CONCLUSION

The goal of our chatbot system is to help the students to guide with the correct source of information in less time.It is advantageous for new applicants for solving queries such as fee structure,types of courses available,etc. For students it is helpful in providing up-to date attendance,result ,notice and many other activities. Students can get the information at their fingertip rather than visiting college.The main motive of our project is to reduce the workload on the college staff and reduce the response time to a student queries.

BIBLIOGRAPHY

1. Ayushi Turkar, Bhvuana k. Zeba Ali Syeda, Ameena Kouser, Ravi P. , "Chatbot for College Using Google Actions", 2019, Pages 268-272.

2. Depti Magnale, Mayur Pawar, Kedar Basanwar, Primal Yadav, Mansi Bhosale, "RUBAN COLLEGE ENQUIRY CHATBOT", International Research Journal of Modernization In Engineering Technology And Educational Institution ", Page 1-5

3. M. Dahiya, "A Tool of Conversation: Chatb", JCSE International Journal of Computer Science & Engineering, 2017, Page 158-161.

4. K. Shingte, Anuja Aditeepatill, Anushreehaudhari, Sharmishta Desai,"Research Paper On Chatbot Development for Educational Institutie", Page 1-5.

5. Shreyashkar Sharma, "CHATBOT DEVELOPMENT USING PYTHON", International Journal of Creative Research Thoughts (IJCRT) July 2020 Pages 5286 - 5289.

6. Payal Jain, "College Enquiry Chatbot Iterative Model", International Journal of Scientific Engineering And Research (IJSER), January 2019, Page 80-83.

7. Donn Emmanuel Gonda, Yiu-Lun Wong, Jing Luo, Chi-Un Lei, "Evaluation of Development Educational Chatbots Based On The Seven Principle For Good Teaching", IEEE International Conference On Teaching, Assessment, And Learning For Engineering (TALE) December 2018, Page 446-453.

8. Nidhi Sarma, Gayatri, "College Enquiry Chatbot", International Research Journal of Engineering & Technology (IRJET), July 2021 Page 1338-1342