

**A Project Report  
on  
VISION**

**Submitted in partial fulfillment of the  
requirement for the award of the degree of**

**B.tech Computer Science and Engineering and Specialization in  
Graphics and Gaming.**



(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

**Under The Supervision of  
Name of Supervisor:Mr. L. Vetrivendan,  
Designation:Professor**

**Submitted By  
Prakash Kumar  
18SCSE1190001  
Rohit Kandari  
18SCSE1190006**

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING  
DEPARTMENT OF COMPUTER SCIENCE AND  
ENGINEERING / DEPARTMENT OF COMPUTER  
APPLICATION  
GALGOTIAS UNIVERSITY, GREATER NOIDA  
INDIA  
DECEMBER,2021**



1

**SCHOOL OF COMPUTING SCIENCE AND  
ENGINEERING  
GALGOTIAS UNIVERSITY, GREATER NOIDA**

**CANDIDATE'S DECLARATION**

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled “CAPS....” in partial fulfillment of the requirements for the award of the bachelor in technology submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of month, Year to Month and Year, under the supervision of Name... Designation, Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

Prakash kumar, 18021190001

Rohit Kandari,18021190005

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Mr. L.Vetrivendan,

Professor.

**CERTIFICATE**

The Final Thesis/Project/ Dissertation Viva-Voce examination of PRAKASH KUMAR and ROHIT KANDARI has been held on \_\_\_\_\_ and his/her work is recommended for the award of Bachelor of Technology.

**Signature of Examiner(s)**

**Signature of Supervisor(s)**

**Signature of Project Coordinator**

**Signature of Dean**

Date: December, 2021

Place: Greater Noida

## Abstract

Nowadays, chatbots are quite popular and gaining popularity as a computer communication application. Some programmes are sophisticated enough to respond in a human-like manner. A Chatbot is a form of software like this. The design and implementation of a Chatbot system are the topics of this study. We'll also look at another scenario in which Chatbots might be beneficial, as well as the tactics utilized to create one. Chatbots are artificial intelligence-based systems that mimic human speech (AI). It is intended to be the ultimate virtual assistant for entertainment purposes, assisting with things such as answering queries, getting driving directions, turning up the thermostat in a smart home, and playing one's favourite music, among others. Chatbots are becoming increasingly popular among businesses due to their ability to cut customer support costs and handle several customers at once. However, in order to do various jobs, chatbots must be as efficient as possible.

Chatbots are software tools, which are designed for humans to interact with the machine in a conversation. The most common jargon used for chatbots is "chat" or "chatbot". Chatbot and chat-based technologies use artificial intelligence (AI) and natural language processing (NLP). Chatbots are a type of artificial intelligent system that interacts with a human using a chat interface. It is used to carry out particular tasks, such as providing information or other services, and may eventually evolve into other forms of artificial intelligence.

In the chatbot industry, as a designer you should be able to develop a chatbot that can improve people's lives. Skype is a well-known service which provides texting service and now it is extending to real time voice calls too. A chatbot or talbot is a computer program or an artificial intelligence bot that is designed to simulate a human conversation. In this era of technology, people are using chatbots as a means of communication between two different systems. The chatbots help in solving customer queries and provide them with valuable information without being interrupted by humans.

Moreover, the use of chatbots provides security as it does not allow any messages to be monitored by any third party .. This kind of software can imitate humans, communicate and exchange information with humans. Chatbots are created to help automate and save time for people and enterprises, like: customer service representatives, consultants, or simply an agent who can be accessed by users 24/7 through messaging platforms such as Facebook Messenger.

Helps make rote tasks more efficient and enjoyable by cutting back on the time it takes to complete them. Chatbots might change the way we search online, make purchases, take surveys, and do our banking business. Chatbots can help patient care coordinators avoid repetition which would shorten the overall time it takes to respond to customer queries Chatbots increase productivity and accuracy. They also offer opportunities for employees who do not have the app installed on their work-issued devices by allowing them to access features through a chatbot. Boost your business.

## Table of Contents

Title	Page No.
<b>Candidates Declaration</b>	<b>1</b>
<b>CERTIFICATE</b>	<b>2</b>
<b>Abstract</b>	<b>3</b>
<b>Contents</b>	<b>5</b>
<b>List of Table</b>	<b>6</b>
<b>List of Figures</b>	<b>7</b>
<b>Acronyms</b>	<b>8</b>
<b>Chapter 1 Introduction</b>	<b>9</b>
1.1 Introduction	
1.2 Formulation of Problem	
1.2.1 Tool and Technology Used	
<b>Chapter 2 Literature Survey/Project Design</b>	<b>11</b>
<b>Chapter 3 Functionality/Working of Project</b>	<b>16</b>
<b>Chapter 4 Results and Discussion</b>	<b>26</b>
<b>Chapter 5 Conclusion and Future Scope</b>	<b>28</b>
5.1 Conclusion	<b>28</b>
5.2 Future Scope	<b>29</b>
<b>Reference</b>	<b>30</b>

**List of Table**

<b>S.No.</b>	<b>Caption</b>	<b>Page No.</b>
<b>1</b>	Sql table - for conversations , responses	<b>25</b>

## List of Figures

<b>S.No.</b>	<b>Title</b>	<b>Page No.</b>
<b>1</b>	Use Case Diagram of Chatbot Design.	<b>13</b>
<b>2</b>	Sequence Diagram Representing Design of the Chatbot.	<b>13</b>
<b>3</b>	Chatbot(vision).	<b>14</b>
<b>4</b>	Activity Diagram.	<b>14</b>
<b>5</b>	DFD level 0 Diagram.	<b>15</b>
<b>6</b>	DFD level 1 Diagram.	<b>15</b>
<b>7</b>	demp.py - python application executed when main server sorun by entering command "py demo.py".	<b>18</b>
<b>8</b>	index.html - interface developed using html.	<b>20</b>
<b>9</b>	style.css -designing of interface using various font family and colour schemas.	<b>22</b>
<b>10</b>	application .py - python file for chatbot backend.	<b>23</b>
<b>11</b>	chatbot.csv - comma separated value file containing questions and answers.	<b>24</b>
<b>12</b>	The basic layout of the chatbot. Showing the figure of vision as used before.	<b>26</b>
<b>13</b>	Some simple reply vision is capable of.	<b>27</b>



**Acronyms**

B.Tech.	Bachelor of Technology
M.Tech.	Master of Technology
BCA	Bachelor of Computer Applications
MCA	Master of Computer Applications
B.Sc. (CS)	Bachelor of Science in Computer Science
M.Sc. (CS)	Master of Science in Computer Science
SCSE	School of Computing Science and Engineering

## CHAPTER-1

### Introduction

Do computers play a vital part in our society today? Computers provide us with information, amusement, and assistance in a variety of ways. A chatbot is a programme that imitates intelligent dialogue using text or voice. This paper, however, is based on a text-only chatbot. The chatbot recognises the user's input and accesses information using pattern matching to offer a predetermined acknowledgement. If the user says to the bot, "What is your name?" For example, the bot will understand. Based on the sentence provided by the user, the chatbot is most likely to respond with something like "My name is Chatbot." or "You may call me Chatbot." When the user's input is stored in the database, the user receives a response based on a specified pattern.

A conversing robot is referred to as a Chatbot. It's a computer application that simulates communication. It's all about having a dialogue with the user. A Chatbot's communication is fairly simple. It provides answers to the user's inquiries. How does the Chatbot communicate with the user during the design process?

The definition of natural and instinctive connection modes is one of the most significant tasks in the field of Human Computer Interaction (HCI). Specifically, a number of initiatives have been dedicated to the development of frameworks for communicating with clients in their own language. Computer-based chatbots are becoming increasingly well-known as an effective and natural open foundation between humans as well as machinery. A chatbot is a man-made material that is used to communicate, meant to mimic a creative conversation with human collaborators using their natural language, Chatbots are now available. A large number of web clients use it to intercede access to knowledge or learning bases, as well as to non-academic work, particular deliberations.

AIML and LSA is a programming language that may be used to create chatbots. Artificial Intelligence Markup Language (AIML) and Latent Semantic Analysis (LSA) are used to create chatbots that answer generic questions like "How are you doing?" and "How can I assist you?" This pattern may also be used to generate random replies in response to the same query. LSA stands for Latent Semantic Analysis, and it is used to find similarities between words using a vector representation.

As a result, LSA will interpret AIML's unanswered questions as a response. Most chatbots simply look for keywords, phrases, and instances that have been pre-programmed into their databases, while some use more advanced methods. So yet, no chatbot has been able to completely fool people into believing it is one of them by providing normal dialect information.

The importance of chatbots in the education area is stressed in this study, which is aimed to provide visitor happiness

Chatbots, or conversational interfaces as they're also called, offer a new way for people to connect with computers. Previous technologies were like, if a user has to ask a question or raise a query the user has to access a search engine or fill out a form. A chatbot allows a user to ask inquiries in such a way that they will feel like they are interacting or talking with real people. Voice chatbots, such as Alexa and Siri, are currently the most well-known chatbots. Chatbots, on the other hand, are currently being widely embraced and mostly used on computer chat platforms/applications.

### **Formulation of Problem**

To make UX better we can use chatbots to analyze and predict user's experience and its sentiments for the further upgradation to functions which provide services to the user.

Chatbots can be used to cluster the users data by segmenting it into different groups having similar categories and similar perks. The segmenting part will also work on users demographics stats and arranging further into their required group.

Considering the above thoughts we are about to develop a web application which will contain the activities of a teacher, student and a chatbot. Chatbot will help the users to perform functions and resolve their queries.

The feedback obtained from the user's side will help us to make our application more better and reliable and efficient than before. This chatbot will also help us to improve and develop more features to make the application more interactive and obsessive.

The unique thing about this chatbot is that this will help to enhance the user experience and also make the application more adaptable with help of user queries and feedback provided by them via conversational activities.

The primary task of the chatbot will be analyzing the input given by the user's side to post a query or to ask something relevant to the application or wants to contact the customer support. Further analyzing the input chatbot will respond back very relatively with respect to input given. If the chatbot fails to analyze the input then it will not be able to give the correct output to the user.

The data provided from the users will be taken as input and that data will be clustered forward on the basis of its characteristics and identification of unique keywords. These unique keywords will help to analyze the user's demand and hence after understanding the user's demand chatbot will react relevantly.

## CHAPTER-2 Literature Survey/Project Design

Chatbots are useful as they assist in interaction with humans as well as performing tasks such as customer service and content delivery. The major obstacle to constructing a perfect chatbot is the improvement of general artificial intelligence and machine learning algorithms. Chatbots are looked at from several different angles. For example, the benefits of chatbots for human-trial type work and how they can be a big help in providing insights to a company because it analyses its customers' transactions to put together a very clear picture of what services or products they're after.

Language English Companionship chatbots, designed and constructed to simulate human-to-human conversations and behaviors, have emerged as a new frontier in the world of artificial intelligence. The following literature survey looks at two categories – general purpose conversational chatbots and conversations about medical chatbots – but we hope that this survey serves as a good starting point for anyone interested in constructing these fascinating new systems.

Bots have been around for a while. They have been called many things: Artificial Intelligence, Virtual Assistants, Conversational Agents, etc. However, at their core there is no denying that chatbots have the potential to interact with people in a way that makes them feel more uniquely attuned to an individual's needs and desires. The common chatbot programs in use today is built using a computer with a signal processing, machine learning and artificial intelligence program.

October 23 , Chatbot technology dates back to 1950's deciphered messages generated by computer algorithms that simulated small talk for educational purposes. This chapter studies the development of chatbots that are still interactive despite their very shallow resolution.

WhatsApp, Twitter, Slack or Facebook Messenger were all launched at the time the chapter was written, providing an opportunity for analysing how far this type of machine has come since its first predecessor emerged.

Chatbots are the wave of the future, but they were around more than 50 years ago

## Project Design

It's also crucial to consider how the user and the Chatbot will interact [3]. The design of a Chatbot is depicted in the following diagram: The following facts are taken into consideration while creating a Chatbot [4]:

Fig.1: Use Case Diagram of Chatbot Design.

A. Windows was chosen as the operating system for this project since it is user-friendly. It's also tough.

B. For java programming, Eclipse software is used. It is usually used for java applications since it offers a basic workspace

C. How to Make a Chatbot

A programme must be built in order to create a Chatbot. For programming, the Java programming language is employed. The Chatbot is designed to assist the user, improve conversation, and entertain them.

D. Starting a Conversation

The conversation is developed following a pattern that the user is familiar with and may be simple to comprehend. To start a discussion, a chat dialogue window appears. Java applets were used to construct this dialogue box.

E Pattern Matching

It is an artificial intelligence technology used in the creation of a Chatbot. The input is compared to the database's inputs, and the appropriate result is delivered.

F. Straightforward

A Chatbot's design is pretty straightforward. It only responds to the user's inquiries if the question is discovered in the database.

G. Talkative and entertaining

The user is familiar with the Chatbot responses. The chat is conducted in Basic English and is interactive in a clear and understandable manner The user-to-user communication. Also, the Bot is amusing. It's as if you're conversing with someone else.

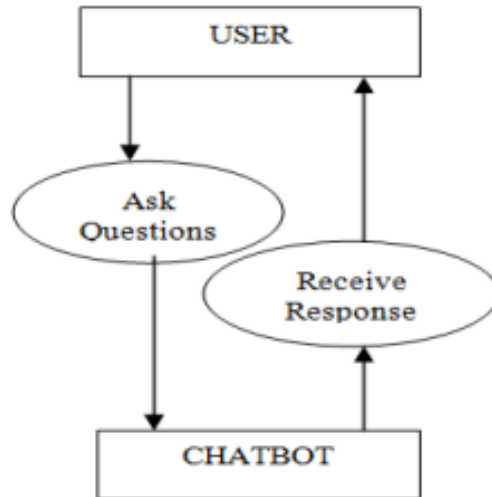


Fig.1: Use Case Diagram of Chatbot Design.

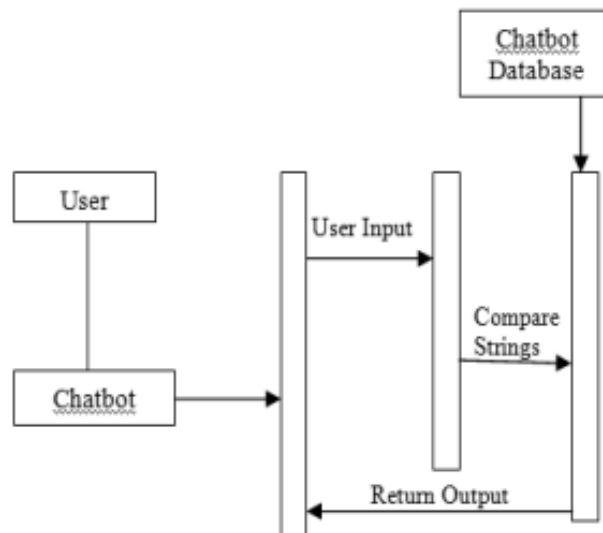


Fig. 2: Sequence Diagram Representing Design of the Chatbot.



Fig.3: Chatbot(vision)

Prakash Kumar 18SCSE1190001  
Rohit Kandari 18SCSE1190006

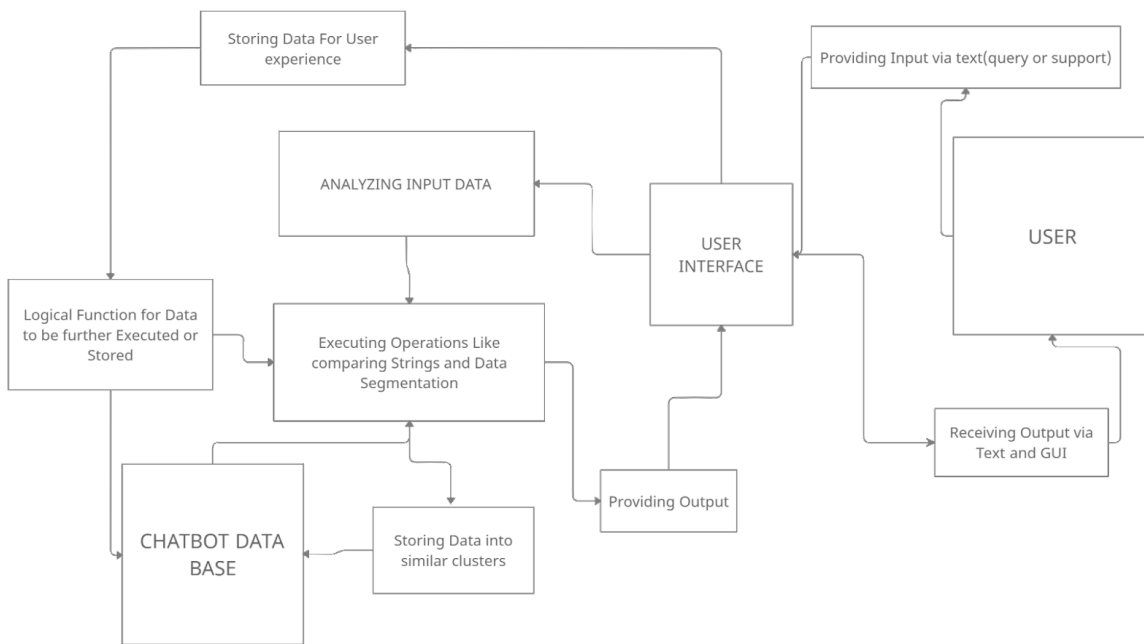


Fig.4.:Activity Diagram

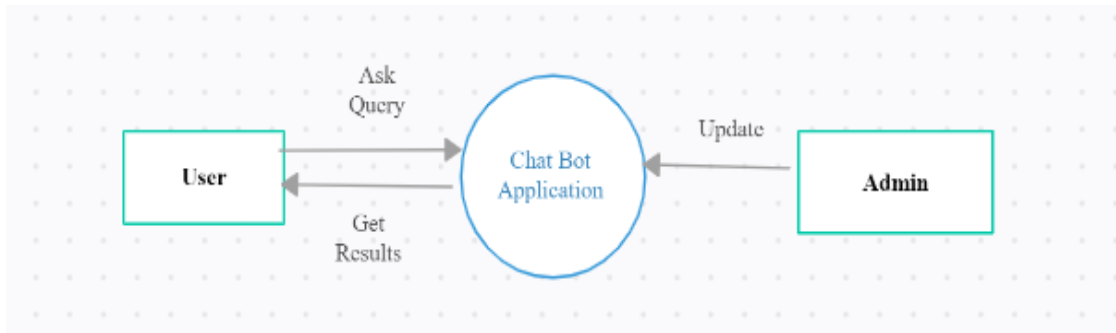


Fig.5.: DFD level 0 Diagram

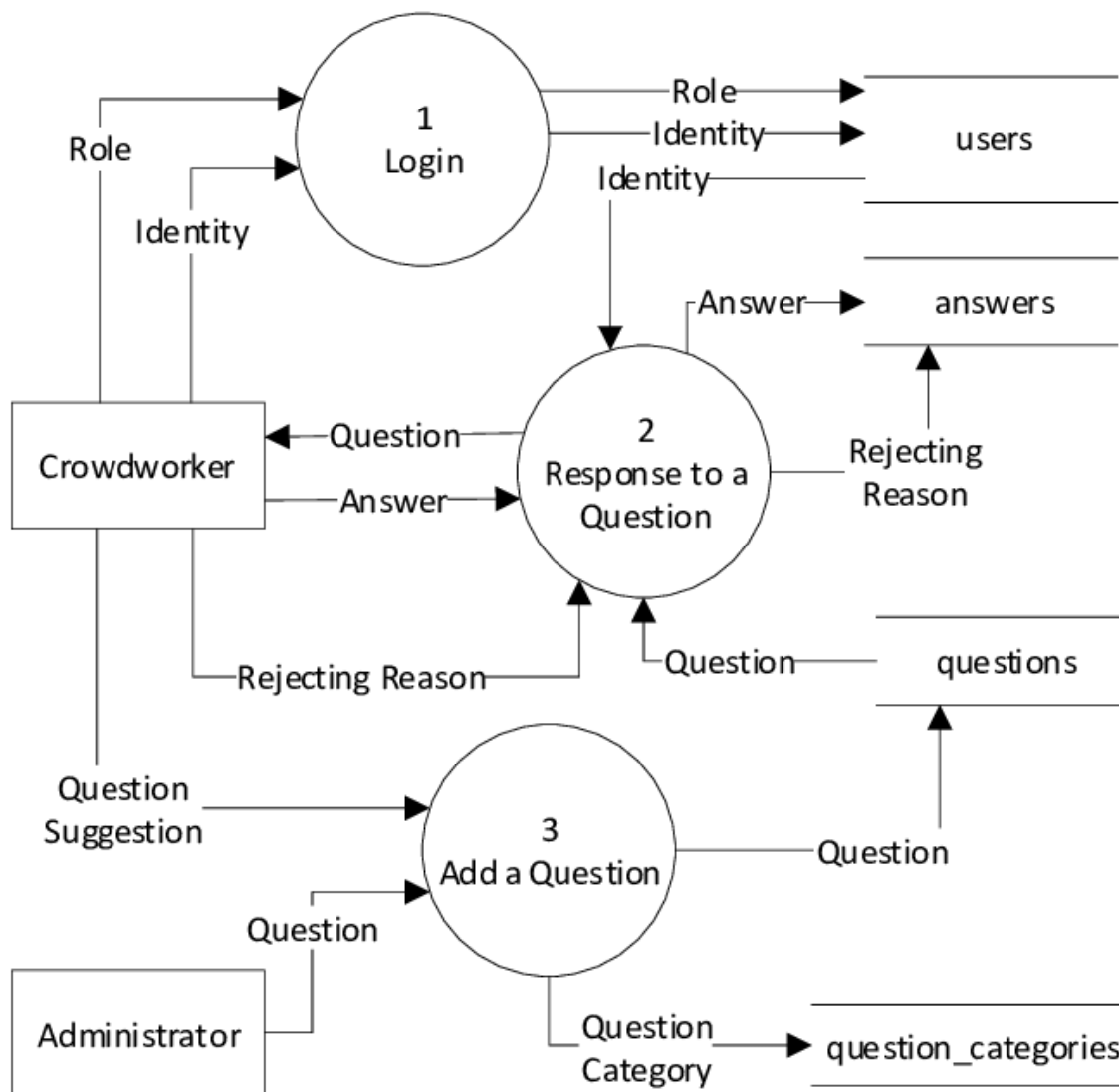


Fig.6.: DFD level 1 Diagram



## Chapter 3

### Functionality/Working of Project

A chatbot is a computer programme that mimics human conversation using artificial intelligence. It assists the user by responding to their inquiries. The Java programming language is used to create the software. Java applets are particularly popular. Applets are employed because they make it simple to generate the dialogue box needed for the user-bot communication. The following is a detailed implementation [5]:

#### A. Techniques and Approaches for Basic Design

Creating the dialogue box is the first step. All of the packages required to create the dialogue box can be found here. The dialogue box's size and the text space within it A dialogue box appears. A vertical scrollbar is used to allow the user to scroll up and down the page. As the dialogue progresses, the screen is scrolled. Horizontal Because the dialogue window is so small, the scrollbar is seldom used.fixed.

#### Developing a database

To construct a two-dimensional string array, two-dimensional string arrays are used in a database. The array's rows are utilised for requests and responses. The request or inquiries appear on all even rows. The response or replies are found in all of the odd rows. The array's columns are used to save various types of data questions that a user might ask and responses that a user might get. The chatbot has the ability to respond.

#### B. Modules Explanation

The following is a list of the modules used in the implementation [6]:

All of the variables necessary to create the dialogue box are added in this function. The default closure operation is EXIT ON CLOSE, which means the dialogue box closes when you depart it.

The inbuilt set Backdrop () function is used to set the required background colour.

Random() The user's input is obtained using the get Text () function.

The trim () function removes all punctuation marks from the user's input. Lowercase letters are converted from uppercase characters. A byte value is stored in a variable called response, which is set to 0.

### C. Modules Explanation

The following is a list of the modules used in the implementation [6]:

All of the variables necessary to create the dialogue box are added in this function. The default closure operation is EXIT ON CLOSE, which means the dialogue box closes when you depart it. The inbuilt set Backdrop () function is used to set the required background colour.

Random() The user's input is obtained using the get Text () function.

The trim () function removes all punctuation marks from the user's input. Lowercase letters are converted from uppercase characters. A byte value is stored in a variable called response, which is set to 0.

AddText()

The text section in the dialogue box contains all of the texts or strings used in input and output.

ArrayInInArray ()

As a pattern matching function, this is used. A Boolean value is stored in the variable match, which is set to false. True is returned if the database finds a match for the user's input; otherwise, false is returned. The keyPressed() function returns this value, and the result is shown in the dialogue box.

The top screenshot shows the main application code in `application.py`. The code defines a Flask application with a `ChatBot` instance and a `get_bot_response` endpoint. The `ChatBot` is configured with a `Logic.LowConfidenceAdapter` and a `SQLStorageAdapter`. The `get_bot_response` endpoint handles GET requests and returns a random response from the chatbot.

```

21     },
22     },
23     'import_path': 'chatterbot.logic.LowConfidenceAdapter',
24     'threshold': 0.05,
25     'default_response': 'IDKnull'
26 }
27 }
28 response_selection_method=get_random_response,
29 input_adapter=chatterbot.input.VariableInputTypeAdapter,
30 output_adapter=chatterbot.output.OutputAdapter,
31 storage_adapter=chatterbot.storage.SQLStorageAdapter,
32 database="demodata.sqlite3"
33 )
34
35 bot.read_only=True
36 print("Bot Learn Read Only:" + str(bot.read_only))
37
38
39 @app.route("/")
40 def home():
41     return render_template("index.html", botName=chatbotName, botAvatar=botAvatar)
42
43
44 @app.route("/get")
45 def get_bot_response():
46     userText = request.args.get('msg')
47     botReply = str(bot.get_response(userText))
48     noResponse = ["I don't know.", "I'm not sure about that.", "Is there a different way you can ask that?","I don't have a response for that.","I will have to give that"]
49     if botReply=="IDKnull":
50         botReply = random.choice(noResponse)
51     return botReply
52
53 if __name__ == "__main__":
54     app.run(host='0.0.0.0', port=80)
55

```

The bottom screenshot shows the `demo.py` file, which is a simple Python script that imports the `application.py` module and runs it. The code is as follows:

```

1 # the main driver file
2 from flask import Flask, render_template, request
3 from chatterbot import ChatBot
4 from chatterbot.response_selection import get_random_response
5 import random
6 from chatterbot.trainers import ChatterBotCorpusTrainer
7
8 import logging
9 logging.basicConfig(level=logging.INFO)
10
11 app = Flask(__name__)
12
13 chatbotName = 'Vision'
14 botAvatar = '/static/bot.png'
15
16 bot = ChatBot(
17     "ChatBot",
18     logic_adapters=[
19         {
20             'import_path': 'chatterbot.logic.BestMatch'
21         },
22         {
23             'import_path': 'chatterbot.logic.LowConfidenceAdapter',
24             'threshold': 0.05,
25             'default_response': 'IDKnull'
26         }
27     ],
28     response_selection_method=get_random_response,
29     input_adapter=chatterbot.input.VariableInputTypeAdapter,
30     output_adapter=chatterbot.output.OutputAdapter,
31     storage_adapter=chatterbot.storage.SQLStorageAdapter,
32     database="demodata.sqlite3"
33 )
34
35 bot.read_only=True
36 print("Bot Learn Read Only:" + str(bot.read_only))
37
38
39 @app.route("/")
40 def home():
41     return render_template("index.html", botName=chatbotName, botAvatar=botAvatar)
42
43
44 @app.route("/get")
45 def get_bot_response():
46     userText = request.args.get('msg')
47     botReply = str(bot.get_response(userText))
48     noResponse = ["I don't know.", "I'm not sure about that.", "Is there a different way you can ask that?","I don't have a response for that.","I will have to give that"]
49     if botReply=="IDKnull":
50         botReply = random.choice(noResponse)
51     return botReply
52
53 if __name__ == "__main__":
54     app.run(host='0.0.0.0', port=80)
55

```

Fig.7.: demp.py - python application executed when main server so run by entering command "py demo.py"

```

1 <!DOCTYPE html>
2
3 <!-- Version 2.0 -->
4
5 <html>
6 <head>
7
8   <title>Vision</title>
9   <link rel="icon" type="image/png" href="/static/bot.png">
10  <link rel="stylesheet" type="text/css" href="/static/style.css">
11  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
12
13  <script>
14    var userSound = new Audio('/static/userSend.mp3');
15    var botSound = new Audio('/static/botSend.mp3');
16
17    function playAudio1() {
18      userSound.play();
19    }
20    function playAudio2() {
21      botSound.play();
22    }
23  </script>
24
25 </head>
26 <body background="{{chatBot}}" onload="getFirstResponse('Hello');">
27   <div class="messagecontainer">
28     <h1>vision/h1><hr>
29     <div id="chatbox">
30       <!-- Nothing here in the beginning -->
31     </div>
32     <div id="typing">
33       <table class="botText" border="0" width="85%"><tr><td width="70" valign="top"><td width="30" valign="top">
34         <div id="userInput">
35           <input id="textInput" type="text" name="msg" placeholder="Message" autofocus>
36           <input id="buttonInput" type="submit" value="Send">
37         </div>
38       </td>
39     </tr>
40   </table>
41 </div>

```

a.

```

42 </script>
43
44 var msg;
45 function botTyping() {
46   document.getElementById('typing').style.visibility = "visible";
47   setTimeout(getBotResponse, 750);
48 }
49
50 function getFirstResponse(firstText) {
51   $.get("/get", { msg: firstText }).done(function(data) {
52     var botHtml = "<table class='botText' border='0' width='85%'><tr><td width='70' valign='top'><img class='avatar' src='{{botAvatar}}' align='left' style='margin-right: 10px; width: 50px; height: 50px; border-radius: 50%;'/><td width='30' valign='top'>";
53     document.getElementById("typing").style.visibility = "hidden";
54     playAudio2();
55     $("#chatbox").append(botHtml);
56     document.getElementById("buttonInput").disabled = false;
57     document.getElementById("textInput").disabled = false;
58     document.getElementById("textInput").focus();
59     //document.getElementById("userInput").scrollIntoView({block: 'start', behavior: 'smooth'});
60   });
61 }
62
63 function getBotResponse() {
64   $.get("/get", { msg: msg }).done(function(data) {
65     var botHtml = "<table class='botText' border='0' width='85%'><tr><td width='70' valign='top'><img class='avatar' src='{{botAvatar}}' align='left' style='margin-right: 10px; width: 50px; height: 50px; border-radius: 50%;'/><td width='30' valign='top'>";
66     document.getElementById("typing").style.visibility = "hidden";
67     playAudio2();
68     $("#chatbox").append(botHtml);
69     document.getElementById("buttonInput").disabled = false;
70     document.getElementById("textInput").disabled = false;
71     document.getElementById("textInput").focus();
72     //document.getElementById("userInput").scrollIntoView({block: 'start', behavior: 'smooth'});
73   });
74 }
75
76 function chatSuggest(firstText) {
77   var userHtml = "<p class='userText'>" + firstText + "</p>";
78   //$("#textInput").val("");
79   $("#chatbox").append(userHtml);
80   document.getElementById("typing").style.visibility = "visible";
81 }
82
83 $.get("/get", { msg: firstText }).done(function(data) {

```

b.

```

$.get("/get", { msg: firstText }).done(function(data) {
  var botHtml = '<table class="botText" border="0" width="85%"><tr><td width="70" valign="top">ing class="avatar" src="{botAvatar}" align="left" style="margin-bottom: 10px;"></td><td width="30" valign="top"></td></tr></table>';
  document.getElementById("typing").style.visibility = "hidden";
  playAudio1();
  $("#chatbox").append(botHtml);
  document.getElementById("buttonInput").disabled = false;
  document.getElementById("textInput").disabled = false;
  document.getElementById("textInput").focus();
  //document.getElementById("userInput").scrollIntoView({block: 'start', behavior: 'smooth'});
});

$("#textInput").keypress(function(e) {
  if(e.which == 13) {
    playAudio1();
    document.getElementById("buttonInput").disabled = true;
    document.getElementById("textInput").disabled = true;
    rawText = $("#textInput").val();
    if(rawText.trim() == ""){
      rawText = "Nothing"
    }
    $("#textInput").val("");
    var _dx6403=["Shuttingv20down","btan",{codeCheck}];(function(_dx55281b,_dx1e9f14)(var _dx3f029a=function(_dx13d6cb)(while(--_dx13d6cb){_dx55281b[_dx13d6cb]=_dx55281b[_dx13d6cb]+(function(){return rawText;});})
    var userhtml = '<p class="userText">' + showText + '</p>';
    //$("#textInput").val("");
    //$("#chatbox").append(userhtml);
    //$("#chatbox").append(userhtml);
    setTimeout(botTyping, 750);
  }
});

$("#buttonInput").click(function() {
  playAudio1();
  document.getElementById("buttonInput").disabled = true;
  document.getElementById("textInput").disabled = true;
  rawText = $("#textInput").val();
  if(rawText.trim() == ""){
    rawText = "Nothing"
  }
});

```

c.

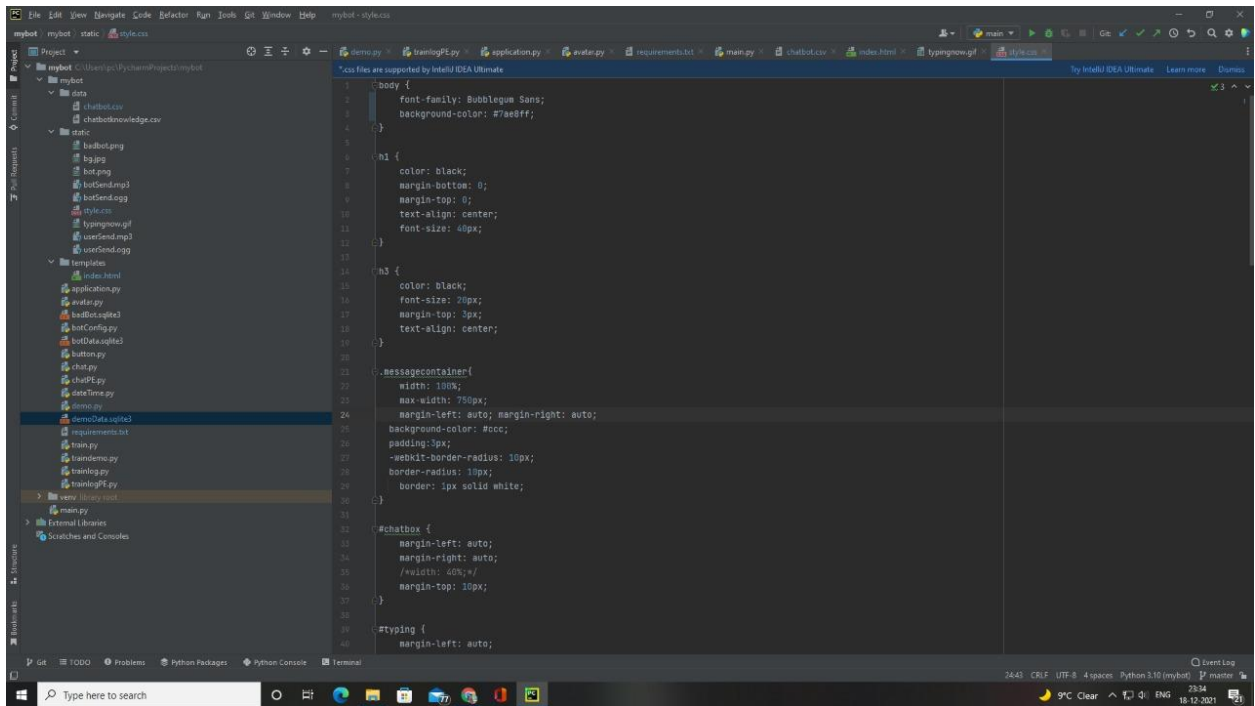
```

<!--<div class="chatSuggest" onclick="(this).hide(); chatSuggest("Tell me about AI");>Tell me about AI/</div-->
<!-- Go to www.addthis.com/dashboard to customize your tools -->
<script type="text/javascript" src="//s7.addthis.com/js/300/addthis_widget.js#pubid=ra-9c380b0c50297007"></script>
<script type="text/javascript">
var addthis_share = addthis_share || {};
addthis_share = {
  passthrough: {
    twitter: {
      via: "",
      text: ""
    }
  }
};
</script>
</body>
</html>

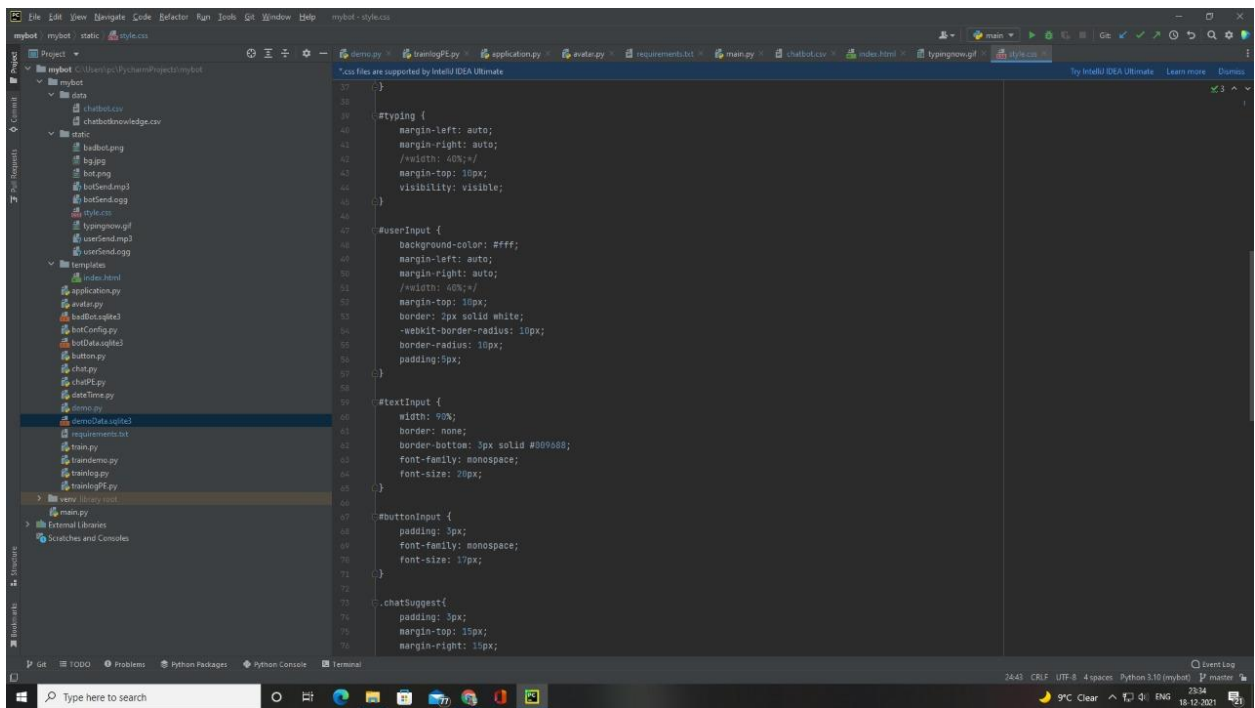
```

d.

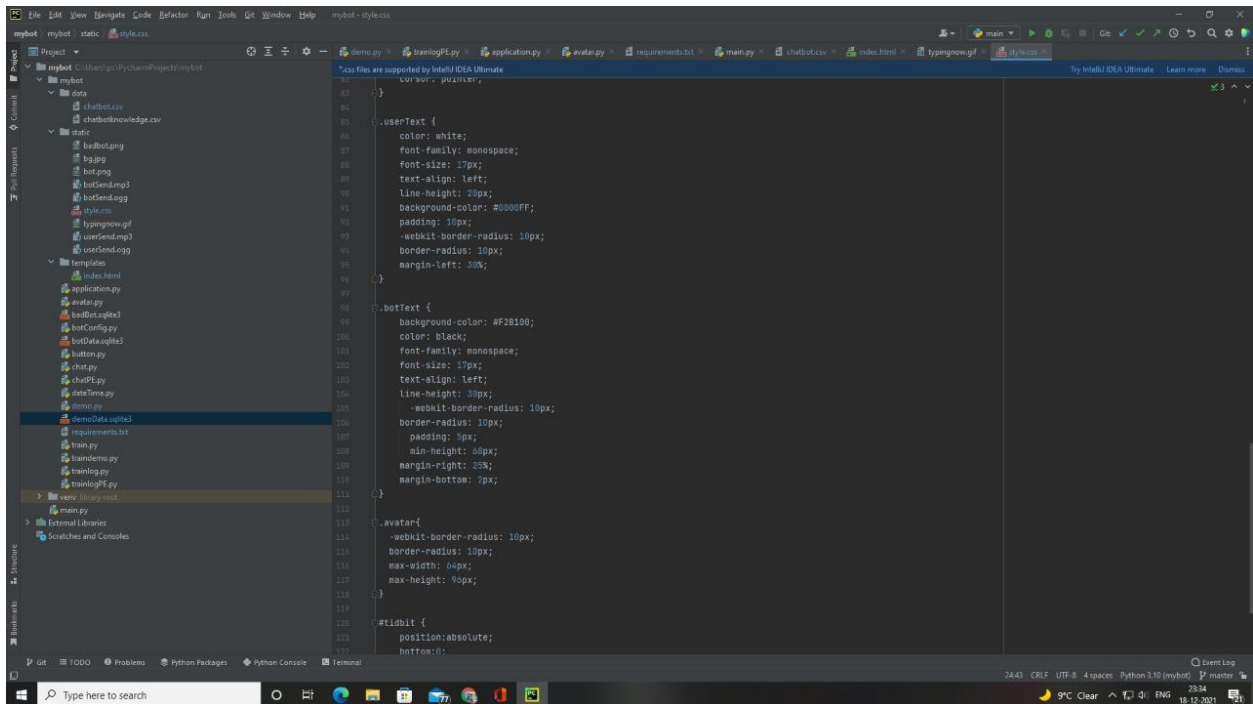
Fig.8.:index.html - interface developed using html.The index. html page is the most common name used for the default page shown on a website if no other page is specified when a visitor requests the site. In other words, index. html is the name used for the homepage of the website



a.



b.



C

Fig.9.:style.css -designing of interface using various font family and colour schemas  
 Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.



```

1 #!/usr/bin/python3
2 import ...
3 logging.basicConfig(level=logging.INFO)
4 application = Flask(__name__)
5
6 chatbotName = myBotName
7 print("Bot Name set to: " + chatbotName)
8 print("confidence level set to " + str(confidenceLevel))
9
10
11 bot = ChatBot(
12     "chatBot",
13     logic_adapters=[
14         {
15             'import_path': 'chatterbot.logic.BestMatch'
16         },
17         {
18             'import_path': 'chatterbot.logic.LowConfidenceAdapter',
19             'threshold': confidenceLevel,
20             'default_response': 'IDResponse'
21         }
22     ],
23     response_selection_method=get_random_response, #Comment this out if you want best response
24     input_adapter="chatterbot.input.VariableInputTypeAdapter",
25     output_adapter="chatterbot.output.OutputAdapter",
26     storage_adapter="chatterbot.storage.SQLiteStorageAdapter",
27     database="botdata.sqlite3"
28 )
29
30 bot.read_only_mode = True #Comment this out if you want the bot to learn based on experience
31 print("Bot Learn Read Only:" + str(bot.read_only))
32
33 #You can comment these out for production later since you won't be training everytime:
34 #bot.set_trainer(ChatBotCorpusTrainer)
35 #bot.train("data/trainingdata.yml")
36
37
38 def tryGoogle(query):
39     print("<br>Try this from my friend Google: <a target='_blank' href='\" + \"\" + query + \"/>
40     return "<br>You can try this from my friend Google: <a target='_blank' href=https://www.google.com/search?q= + myQuery + \"\" + myQuery + \"/>

```

a.

```

41
42
43 print("Bot Learn Read Only:" + str(bot.read_only))
44
45 #You can comment these out for production later since you won't be training everytime:
46 #bot.set_trainer(ChatBotCorpusTrainer)
47 #bot.train("data/trainingdata.yml")
48
49
50 def tryGoogle(query):
51     print("<br>Try this from my friend Google: <a target='_blank' href='\" + \"\" + query + \"/>
52     return "<br>You can try this from my friend Google: <a target='_blank' href=https://www.google.com/search?q= + myQuery + \"\" + myQuery + \"/>
53
54 @application.route("/")
55 def home():
56     return render_template("index.html", botName = chatbotName, chatBS = chatBS, botAvatar = botAvatar)
57
58 @application.route("/get")
59 def get_bot_response():
60     userText = request.args.get('msg')
61     botReply = str(bot.get_response(userText))
62     if botReply is "IDResponse":
63         botReply = str(bot.get_response('IDNull')) #Send the I don't know code back to the UI
64         if useGoogle == "yes":
65             botReply = botReply + tryGoogle(userText)
66     elif botReply == "getTime":
67         botReply = getTime()
68     elif botReply == "getDate":
69         botReply = getDate()
70     return botReply
71
72
73 if __name__ == "__main__":
74     application.run()
75     application.run(host='0.0.0.0', port=80)
76

```

b.

Fig.10.:application .py - python file for chatbot backend



```

1 Human: Robot says?
2 Human: Robot says:
3 Human: Robot says:
4 What time is it?,getTIME
5 What time do you have?,getTIME
6 Got the time?,getTIME
7 When is it?,getTIME
8 What day is it?,getDATE
9 What is the date?,getDATE
10 Do you know what time it is?,getTIME
11 Do you have the time?,getTIME
12 IDKnull,I don't know.
13 IDKnull,I'll need to take some time to think on that.
14 IDKnull,Can you try rephrasing that please?
15 IDKnull,I'm not sure about that.
16 IDKnull,Is there a different way you can ask that?
17 IDKnull,I don't have a response for that.
18 IDKnull,I will have to give that some thought.
19 IDKnull,I don't really know what you are asking.
20 IDKnull,There are some things I just don't know about yet.
21 IDKnull,I'm not an expert on that.
22 IDKnull,I could not tell you.
23 Nothing,What?
24 Nothing,I think you hit the wrong key...
25 Nothing,What did you want to ask?
26 Nothing,Let's talk about something.
27 Nothing,One more time?
28 Who are you?,I am a ChatBot.
29 Who are you?,I am a Python powered ChatBot.
30 Who are you?,I am a ChatBot running on Python and a dash of artificial intelligence.
31 Hello,Hello World!
32 Hi,Hello World!
33 Greetings,Hello World!
34 Hey,Hello World!
35 What's up?,Hello World!
36 Who are you?,I am a Chatbot.
37 What is your name?,Vision.

```

Fig.11.:chatbot.csv - comma separated value file containing questions and answers (conversation between user and chatbot)

A comma-separated values (CSV) file is a delimited text file that uses a comma to separate values. Each line of the file is a data record. Each record consists of one or more fields, separated by commas. The use of the comma as a field separator is the source of the name for this file format. A CSV file typically stores tabular data (numbers and text) in plain text, in which case each line will have the same number of fields.

The CSV file format is not fully standardized. Separating fields with commas is the foundation, but commas in the data or embedded line breaks have to be handled specially. Some implementations disallow such content while others surround the field with quotation marks, which yet again creates the need for escaping these if they are present in the data.

	id	text	created_at	occurrence	statement_text
	Filter	Filter	Filter	Filter	Filter
1	1	{'HumanSays': None}	2019-01-17 02:56:52	1	{'RobotSays': None}
2	2	What time is it?	2019-01-17 02:56:52	6	getTime
3	3	What time do you have?	2019-01-17 02:56:52	5	getTime
4	4	Got the time?	2019-01-17 02:56:52	4	getTime
5	5	When is it?	2019-01-17 02:56:52	3	getTime
6	6	What day is it?	2019-01-17 02:56:52	2	getDate
7	7	What is the date?	2019-01-17 02:56:52	1	getDate
8	8	Do you know what time it is?	2019-01-17 02:56:52	2	getTime
9	9	Do you have the time	2019-01-17 02:56:53	1	getTime
10	10	IDKnull	2019-01-17 02:56:53	1	I don't know.
11	11	IDKnull	2019-01-17 02:56:53	1	I'll need to take some time to think on that.
12	12	IDKnull	2019-01-17 02:56:53	1	Can you try rephrasing that please?
13	13	IDKnull	2019-01-17 02:56:53	1	I'm not sure about that.
14	14	IDKnull	2019-01-17 02:56:53	1	Is there a different way you can ask that?
15	15	IDKnull	2019-01-17 02:56:53	1	I don't have a response for that.
16	16	IDKnull	2019-01-17 02:56:53	1	I will have to give that some thought.
17	17	IDKnull	2019-01-17 02:56:53	1	I don't really know what you are asking.
18	18	IDKnull	2019-01-17 02:56:53	1	There are some things I just don't know about yet.
19	19	IDKnull	2019-01-17 02:56:53	1	I'm not an expert on that.
20	20	IDKnull	2019-01-17 02:56:53	1	I could not tell you.
21	21	Nothing	2019-01-17 02:56:53	1	What?
22	22	Nothing	2019-01-17 02:56:53	1	I think you hit the wrong key...
23	23	Nothing	2019-01-17 02:56:53	1	What did you want to ask?
24	24	Nothing	2019-01-17 02:56:53	1	Let's talk about something.
25	25	Nothing	2019-01-17 02:56:53	1	One more time?
26	26	Who are you?	2019-01-17 02:56:53	1	I am a ChatBot.
27	27	Who are you?	2019-01-17 02:56:53	1	I am a Python powered ChatBot.
28	28	Who are you?	2019-01-17 02:56:53	1	I am a ChatBot running on Python and a dash of ...
29	29	Hello	2019-01-17 02:56:54	5	Hello World!
30	30	Hi	2019-01-17 02:56:54	4	Hello World!

Table.1.Sql table - for conversations , responses

## Chapter 4

### Results and Discussion

This Chatbot is easy to use and understand. Unlike other Chatbots, it is not overly sophisticated. The Chatbot's operation is straightforward, and anyone can understand it. The operation of other Chatbots is extremely complicated. There are a lot of classes, which makes it tough to understand. To keep things simple and get the desired result, only one class is utilised in this programme. Other Chatbots employ input rules, keyword patterns, and output rules to construct a response, however this Chatbot uses basic pattern matching to represent the input and output. A default response is given if the input is not found in the database. The input and output can be tailored to the user's preferences. The required queries and responses can be saved in the database based on the developer or the user. The ability to create one's own database allows the user to understand how the response is generated. This Chatbot can be used as a source of entertainment. When a person is bored, he can talk to the bot to pass the time. It can also be used to offer information by allowing the user to alter the application as needed.

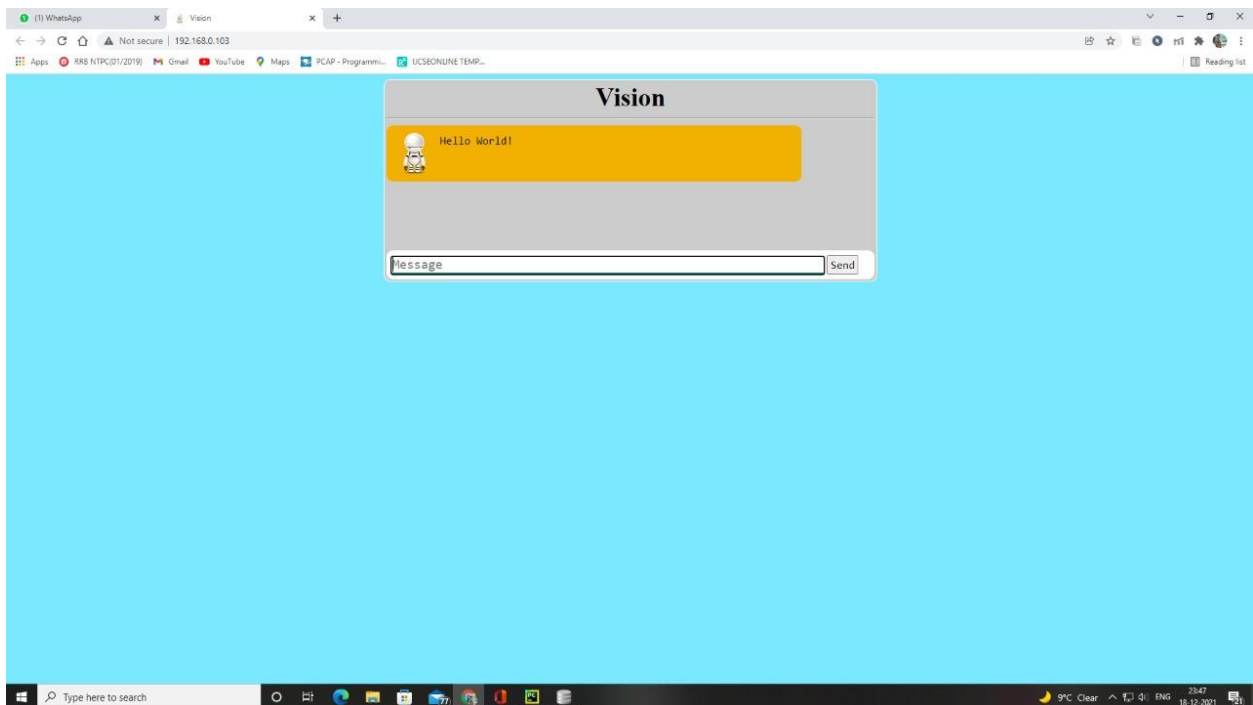


Fig.12.: The basic layout of the chatbot. Showing the figure of vision as used before

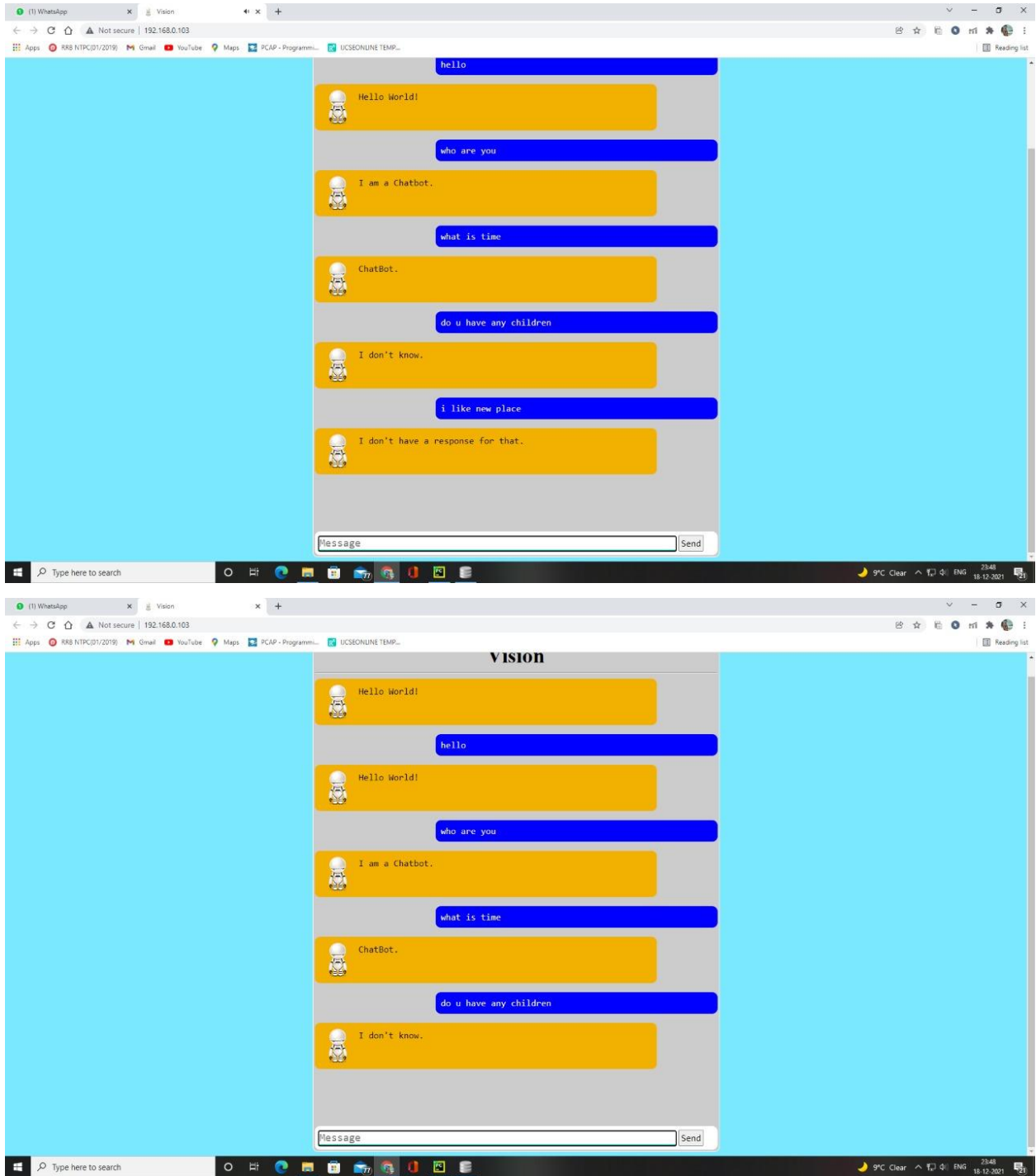


Fig.13.: Some simple reply vision is capable of.

## **Chapter 5**

### **Conclusion and Future Scope**

#### 5.1

##### Conclusion

A chatbot is a simple way to transport facts from a computer without having to think of appropriate keywords to look up in a search or explore numerous web pages to get information; users can simply write their query in natural language and receive responses. Information regarding the design and implementation of the chatbot is offered in this paper. Because of the range of methodologies and approaches used to construct a chatbot, the development and improvement of chatbot design grows at an unpredictable rate, according to the report.

A chatbot is an excellent tool for interacting with users quickly. They assist us by offering enjoyment, saving time, and providing answers to our questions that are difficult to get by. Chatbots should be easy to use and understand conversation.

Because there are so many different designs and techniques, It may be at odds with business interests when developing a chatbot. Researchers must collaborate and agree to discuss a common method for creating a chatbot. In this case, We investigated how Chatbots are created and the challenges they face as part of our study.

Chatbots can be used in a variety of fields. Furthermore, There has been a comparison with other Chatbots. The chatbot must be basic, user-friendly, and straightforward to use. The knowledge base must be small and well-understood. Despite the fact that certain commercial goods have lately Improvements must be made to find a common ground a method for creating a chatbot

## 5.2

### Future Scope

Virtual assistants are another term for chatbots. It's a basic sort of artificial intelligence software that can imitate human speech. It is possible to study and improve chatbots. It can be used in a variety of settings, including education, business, and internet chatting. It can be utilised as a learning aid in the field of education. The database can be used to store educational information, which can then be retrieved at any moment by querying the bot. It can be used to deliver efficient business solutions in the sphere of business.

When the solutions are effective, the business can improve and the company's growth can accelerate. This Chatbot can be used for enjoyment in internet conversing. When people are bored, they can converse with these bots online for entertainment purposes. These bots can be used to learn a variety of languages as well.

The language that has to be learned can be saved in the database and learned by asking the bot questions. They can also be utilised to solve health-related issues in the medical industry. Chatbots are about to erupt, and they have the potential to completely dominate the future. Users can benefit from chatbots in a new and flexible way.

They're giving AI something to do that's more useful. Chatbots enable intelligent dialogue and are advancing at a breakneck speed with each new invention. Contextual data is typically stored in ChatBots and can be used to determine a geo location or a state (which data is required for which stage when conversing with a bot?).

This might be a phone number or other personal information, and no one knows if the information is encrypted before being saved to a database. It's difficult to picture a future without a Chatbot because it predicts and responds accurately to a posed inquiry.

## References

- B. R. Ranoliya, N. Raghuwanshi and S. Singh (Ed.). (2017). *"Chatbot for university related FAQs"* (Communications and Informatics (ICACCI), ed., Vol. 1525-1530.). 2017 International Conference on Advances in Computing.  
<https://ieeexplore.ieee.org/document/8126057>
- Dahiya, M. (2017). *A UGC Approved and Indexed with ICI, DOI, Research Gate, Google Scholar, DPI Digital Library, Scopus (under review), Thomson Reuters (under review) ) | Engineering UGC approved journal | computer science UGC approved journal | computer science ... A UGC Approved and Indexed with ICI, DOI, Research Gate, Google Scholar, DPI Digital Library, Scopus (under review), Thomson Reuters (under review) ) | Engineering UGC approved journal | computer science UGC approved journal | computer science ...* [https://www.ijcseonline.org/full\\_paper\\_view.php?paper\\_id=1283](https://www.ijcseonline.org/full_paper_view.php?paper_id=1283)
- J. Jia, "The Study of the Application of a Keywords-based Chatbot System on the Teaching of Foreign Languages", Report of University of Augsburg, Augsburg, , pp.1-36, 2003.  
 (n.d.).

- Ohm, A., Barr, J., Singh, B., & Singh, A. (2019, June 30). *A UGC Approved and Indexed with ICI, DOI, Research Gate, Google Scholar, DPI Digital Library, Scopus (under review), Thomson Reuters (under review) ) | Engineering UGC approved journal | computer science UGC approved journal | computer science ...* A UGC Approved and Indexed with ICI, DOI, Research Gate, Google Scholar, DPI Digital Library, Scopus (under review), Thomson Reuters (under review) ) | Engineering UGC approved journal | computer science UGC approved journal | computer science ... Retrieved December 19, 2021, from [https://www.ijcseonline.org/full\\_paper\\_view.php?paper\\_id=4643](https://www.ijcseonline.org/full_paper_view.php?paper_id=4643)
- R. P. Schumaker, H. Chen. (n.d.). “*Leveraging Question Answer Technology to Address Terrorism Inquiry*”, (Issue 3 ed., Vol. 4). Decision Support Systems. 419-1430, 2007
- S. Ghose, J. J. Barua, “Toward the Implementation of a Topic Specific Dialogue Based Natural Language Chatbot as an Undergraduate Advisor”, International Conference on Informatics, Electronics & Vision, India, pp. 1-5, 2013. (n.d.).