AUTOBOTS

A PROJECT REPORT

Submitted by

DAYESH RAVAL
(20BAI10189)
SHUBHAM GUPTA
(20BAI10359)
SHIVIKA BANSAL
(20BAI10361)
SAKSHI JAISWAL
(20BAI10102)

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

SCHOOL OF COMPUTING SCIENCE AND TECHNOLOGY WITH SPECIALIZATION IN ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

VIT BHOPAL UNIVERSITY

KOTHRIKALAN, SEHORE

MADHYA PRADESH - 466114

DEC 2021

VIT BHOPAL UNIVERSITY, KOTHRIKALAN, SEHORE
MADHYA PRADESH – 466114

BONAFIDE CERTIFICATE

Certified that this project report titled **“AUTOBOTS”** is the bonafide work of **DAYESH RAVAL. (Register No :20BAI10189), SHUBHAM GUPTA (Register No :20BAI10359), SHIVIKA BANSAL (Register No :20BAI10361), SAKSHI JAISWAL (Register No :20BAI10102),”** who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported at this time does not form part of any other project/research work based on which a degree or award was conferred on an earlier occasion on this or any other candidate.

PROGRAM CHAIR

Dr. S Sountharrajan, Department Head

School of Computer Science and Engineering

VIT BHOPAL UNIVERSITY

PROJECT GUIDE

Dr. S Suthir, Faculty

School of Computer Science and Engineering

VIT BHOPAL UNIVERSITY

The Project Exhibition I Examination is held on 21/12/2021.

ACKNOWLEDGEMENT

First and foremost, I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to **Dr S Sountharrajan**, Head of the Department, School of Aeronautical Science for much of his valuable support encouragement in carrying out this work.

I would like to thank my internal guide **Dr. S Suthir**, for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Computer Science and Engineering, who extended directly or indirectly all support.

Last, but not least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ABBREVIATIONS	MEANING	PAGE NO.
IDE	Integrated Development Environment	VI
FAQ’s	Frequently Asked Questions	4
AI	Artificial Intelligence	2
NLP	Natural Language Processing	2
ML	Machine Learning	2
AIML	Artificial Intelligent Markup Language	5
CSS	Cascading Style Sheets	17
URL	Uniform Resource Locator	14
UI	User Interface	18
CL	Command Line	16
GUI	Graphical User Interface	18
MDI	Menu-Driven Interface	18
FBI	Form-Based Interface	18
NLI	Natural Language Interface	18

LIST OF FIGURES AND GRAPHS

LIST OF TABLES

ABSTRACT

- **Purpose**

Nowadays, almost 30 percent of the tasks are fulfilled by chatbots. Companies use the chatbots to provide services like customer support, generating information, etc. Our chatbot, **NeoBot**, an artificial agent, flask chatterbot etc. is basically software program driven by artificial intelligence which serves the purpose of making a conversation with the user by texts or by speech. This flask chatterbot, **Neobot** is inclined towards performing a specific task for the user. Normally, any chatbot is merely a computer program that fundamentally simulates human conversations.

- **Methodology**

  The program selects the closest matching response from the closest matching statement that matches the input. It then chooses the response from the known selection of statements for that response. Our chatbot, **Neobot** uses rule-based approach i.e., it is trained according to rules. Based on this, it answers simple queries. The design of the Chatbot is to be taken place on the chatterbot framework. The project will include coding of the app(app.py), code of the template of the app (index.html) and style of the CSS sheet template (style.css), which will take place on the Python IDE- PYCHARM.

- ## Findings

We found many chatbots that would serve the purpose of the users but, the two main principles that our chatbot, **Neobot** would guarantee for appropriate interaction between humans and computer are Visibility and affordance. The users can start a conversation where they wish to begin with, but it is not feasible to immediately respond as the logs of chats are stored and referred to in the future. Our chatbot **NeoBot** can interact with clients, guide them to the right information and answer queries. As a result, it can help ease the workload of advisors. Meanwhile, customers can find solutions faster, boosting satisfaction with answering FAQ's and 24/7 support. Chatbot are effective tools when it comes to education, e-commerce etc. The main aim of our chatbot, **Neobot** is to help people, facilitate their work, and to interact with computer using natural language

CHAPTER-1:

PROJECT DESCRIPTION AND OUTLINE

1.1) Introduction

In 1994, when Michael Mauldin produced his first chatbot and called it “Julia” and that’s the time when the word “chatterbot” entered first time in our dictionary. A chatbot is described as a computer program designed to simulate conversation with human users, particularly over the internet. It is software designed to mimic how people interact with each other. It can be seen as a virtual assistant that interacts with users through text messages or voice messages and this allows companies to get closer to their customers.

There are many Chatbot platforms that help companies to create personalized chatbots. The database and APIs play a big role in these chatbot platforms. Some of those platforms are Amazon Lex, Microsoft Azure Bot, Chatterbot. A chatbot is programmed to work independently from a human operator. It can answer questions formulated to it in natural language and respond like a real person.

1.2) **Motivation for the work**

As the world is becoming even more interconnected and digitized, chatbots are becoming customers' and companies' best allies. Chatbots are conversational robots programmed and designed to instantly answer users' questions. For instance, the robot to whom you ask questions about a product in the tiny popup window at the bottom-right corner of your screen when doing online shopping, is a chatbot. Nevertheless, chatbots have many more use cases and are very versatile. They can be used in Human Resources management, in schools' administrations and in many other contexts. By using algorithms, some basic chatbots, called rules-based chatbots, can handle simple tasks, like suggesting products or making appointments. However, some very high-performance chatbots, called Artificial Intelligence (AI) chatbots, use Natural Language Processing (NLP) and Machine Learning (ML) to perfect their understanding of users' inputs. These chatbots can deal with far more complex tasks, which we have tried to achieve through our chatbot.

1.3) About Introduction to the project including techniques

Implementing our chatbot technology, we would be bringing about a diverse range of benefits. We would be able to interact with clients, guide them to the right information and answer queries. Every time the chatbot gets the input from the user, it saves the input and the response which helps the chatbot with no initial knowledge to evolve using the collected responses. The program selects the closest matching response from the closest matching statement that matches the input.it then chooses the response from the known selection of statements for that response. This chatbot will have a textbox where we can give the user input, and the bot will generate a response for that statement.

1.5) Problem Statement

From literature, we found out that in general, chatbot are functions like a typical search engine. Although chatbot just produced only one output instead of multiple outputs/results, the basic process flow is the same where each time an input is entered, the new search will be done. This research is focused on enabling our AI-powered interface, as it can stimulate human conversation and engage with customers. In chatbot context, this functionality will enhance the capability of chatbot's input processing. In attempt to augment the traditional mechanism of chatbot processes, we used the relational database model approach to redesign the architecture of chatbot in a whole as well as incorporated the Extension and Prerequisite codes (our proposed codes). Extension and prerequisite enabled relations between responses that significantly make it easier for user to chat with chatbot using the same approach as chatting with an actual human.

1.6) Objective of the work

Our AI-powered interface, **Neobot** can stimulate human conversation and engage with customers. It can interact with clients, guide them to the right information and answer queries. As a result, it can help ease the workload of advisors. Meanwhile, customers can find solutions faster, boosting satisfaction with answering FAQ’s and 24/7 support. Also, sales teams can use our chatbot to generate leads. Overall, the customer service experience is smoother, and life becomes easier for users and clients alike.

1.7) Organization of the project

- Installation of the required module: chatterbot library.
- Addition of two directories: static and templates for HTML AND CSS files.
- Setting up the flask chatterbot app (app.py).
- After setting up, complete the coding of the template of the app and style sheet of the CSS file.
- Execute the app.py code and check the terminal for an address shown.
- Click the address on the output, and you will get the app with the chatbot in the browser.

1.8) Summary

AIML has been used for creating chatbots. Artificial Intelligence Markup Language (AIML) are used for developing chatbots, which are used to define general pattern-based queries. This pattern can also be used to give random responses for the same query in the chatbot. The next step towards building chatbots involves helping people to facilitate their work and interact with computers using natural language or using their set of rules. Future Such chatbots, backed by machine-learning technology, will be able to remember past conversations and learn from them to answer new ones.

CHAPTER-2:

RELATED WORK INVESTIGATION

2.1) Introduction

As we begin down the path of creating a chat-based & utilitarian mobile experience for museums, it's important to note that we on the Studio team do not consider ourselves to be experts in chatbots, artificial intelligence or conversational interfaces. Acknowledging this, our primary goal right now is to learn as much as possible (as quickly as possible) from the experts that do exist within the bot domain. We're currently in deep with the first phase of this process — Formative Research and Evaluation — which is designed in-part to rapidly spin up the team's knowledge base and understanding of the conversational experience landscape.

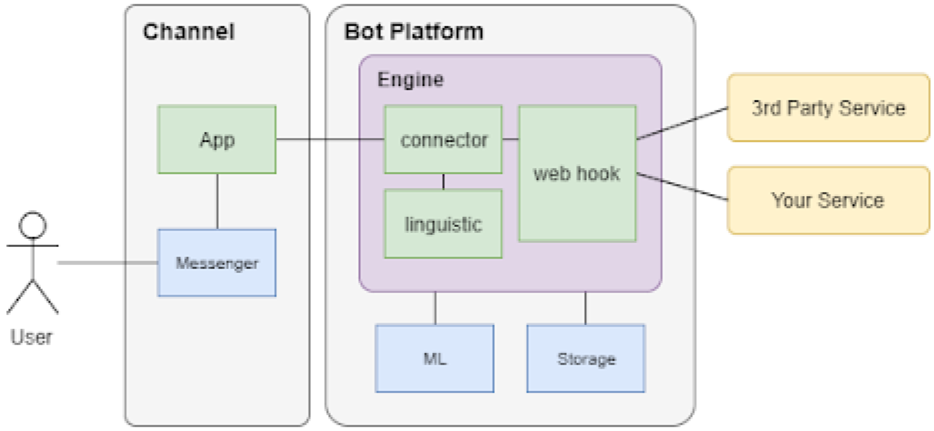
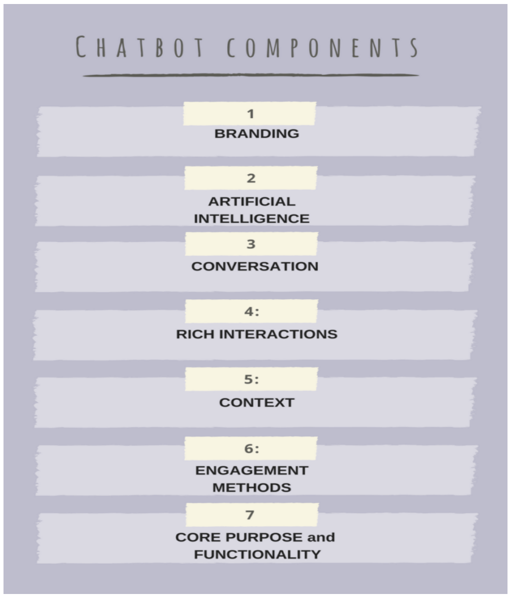


FIGURE-1

2.2) Core Area of the project

New technologies are always exciting. They bring a wide map of opportunities to develop human-computer interactions. Conversational interfaces, also called chatbots, are a great example of this in action. Chatbots are software applications that recreate conversations between humans and internet services. These conversational interfaces work with a set of structured and repetitive tasks to simulate a fluent conversation. They have limited mental processes and work with linear answering data. However, their functionality can be improved by implementing artificial intelligence. Using AI enables chatbots to predict human conversation, understand sentiments and recognize images and emotions in the conversation.



CHATBOT COMPONENTS

1
BRANDING

2
ARTIFICIAL INTELLIGENCE

3
CONVERSATION

4:
RICH INTERACTIONS

5:
CONTEXT

6:
ENGAGEMENT METHODS

7
CORE PURPOSE and FUNCTIONALITY

FIGURE-2

2.3) **Existing Approaches/ Methods**

- Normally, any chatbot is merely a computer program that fundamentally simulates human conversations. It allows a form of interaction between a human and a machine the communication, which happens via messages or voice command.

- A chatbot is programmed to work independently from a human operator. It can answer questions formulated to it in natural language and respond like a real person. It provides responses based on a combination of predefined scripts and machine learning applications.

- When it is asked a question, A chatbot will respond based on the knowledge database available to it at that point in time.

2.4) Pros and cons of the stated approaches

With the increasing advancements, there also comes a point where it becomes difficult to work with chatbots. Following are a few limitations we face with these chatbots are:-

- **Domain Knowledge**– Since true artificial intelligence is still out of reach, it becomes difficult for any chatbot to completely fathom the conversational boundaries when it comes to conversing with a human.

- **Personality** – Not being able to respond correctly in addition to poor comprehension skills, has been the most frequent errors for the chatbots, adding a personality to a chatbot is still a benchmark that seems within our reaches. But we are more than hopeful with the existing innovations and progress-driven approaches.

FIGURE-3

2.5) Issues/Observations from investigation

When it is about talking with a human, randomness is always there and for a program to understand that randomness, you need to know these random possibilities. Which you will always fall short off, it’s like handling the case you don’t know exists. I think once you know the problems and challenges, the solution follows. So, this made me write this post. There are many challenges that you will face while creating your chatbot but these you should consider well in advance to make your users happy.

2.6) Summary

A chatbot is merely a computer program that fundamentally simulates human conversations. A chatbot that functions through AI and machine learning has an artificial neural network inspired by the neural nodes of the human brain. Chatbots are programs that can do talk like human conversations very easily. Chatbots are effective tools when it comes to education, e-commerce etc. The main aim of our chatbot, **Neobot** is to help people, facilitate their work, and to interact with computer using natural language

CHAPTER-3:

REQUIREMENT ARTIFACTS

3.1) Introduction

In this module, we will be covering on how usable is our chatbot, taking a look towards the software and hardware requirements, it's performance, security etc. We will get a better idea on the outlook of our chatbot.

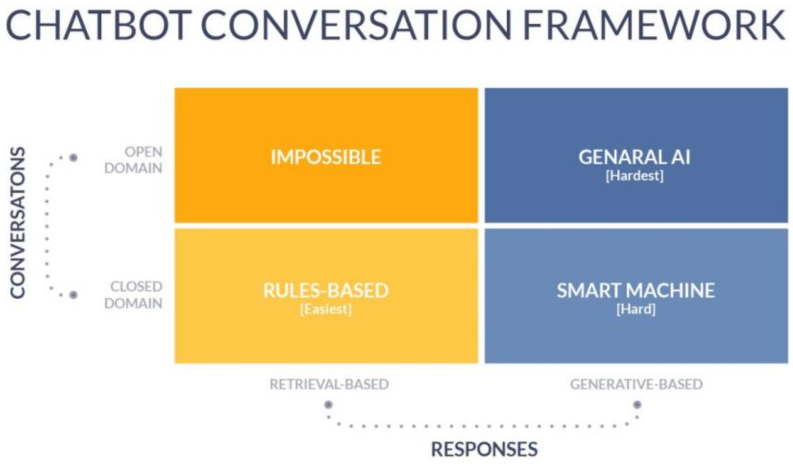


FIGURE-4

3.2) Hardware and Software Requirements

HARDWARE	SOFTWARE
Processors: Intel® core™ i5-8265U is being used (also work for Intel® core™ i3 models)	**Platform**:- Python IDE- PYCHARM, Google collab can also be used.
RAM: 4Gb, 8Gb	**Directories created**:- a) **Static directory** chat interface b) **Templates directory** Don't send empty message c)**README.md** Update README.md d)**app.py** Updated to latest chatterbot major version e) **requirements.txt**
ROM: 476 Gb	Flask, Chatterbot mainframe module(**Chatterbot Corpus**)
No. of cores: 2,4	HTML and CSS files i.e. (**index.html & style.css**) Bootstrap

Table-1

3.3.1) **Performance and Security**

Chatbots are software systems created to interact with humans through chat. The first chatbots were able to create simple conversations based on a complex system of rules. The limits of these systems have been overcome by chatbots that use AI and machine learning to interpret the intents of their interlocutor. Chatbots can help in many practical cases and drastically reduce management costs. There are many examples that have become well-known successful use cases. For example, retailer H&M uses them to guide users through their purchase process on their website. In general, many support systems use chatbots to achieve operational efficiency, including answering common questions or helping users solve repetitive tasks. And some of them are very complex, such as those offering commercial offers or giving advice as a robo-advisor. Chatbots are much like any other digital technology – they are only as secure as you make them. Though there's the potential for them to be used as a backdoor by hackers, if you're willing to invest appropriately they're as safe and secure as any other customer-facing technology.

3.3.2) Look and Feel requirements

- Import and load the data file.

- Preprocess data.

- Create training and testing data.

- Build the model.

- Predict the response

*** The chatbot will be deployed through a URL link on web server or search engines***

3.4) Summary

Chatbot architecture is the heart of chatbot development. Based on the usability and context of business operations the architecture involved in building a chatbot changes dramatically. So, based on client requirements we need to alter different elements; but the basic communication flow remains the same.

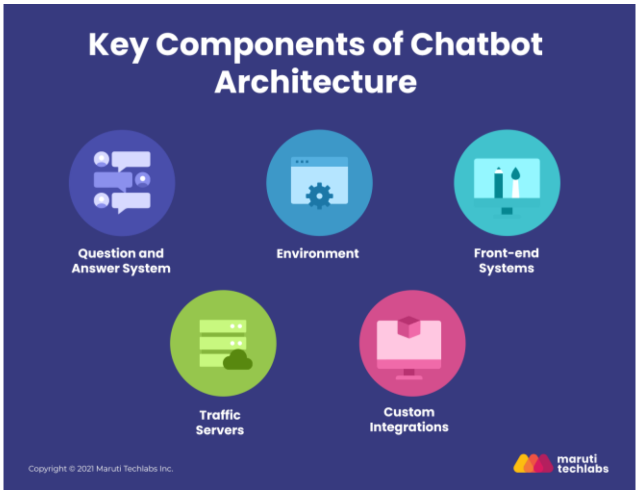

Key Components of Chatbot Architecture
Question and Answer System
Environment
Front-end Systems
Traffic Servers
Custom Integrations
Copyright © 2021 Maruti Techlabs Inc.
maruti techlabs

FIGURE-5

CHAPTER-4:

DESIGN METHODOLOGY AND ITS NOVELTY

4.1) Methodology and goal

Our flask chatterbot makes it easy to create software that engages in conversation.Every time the chatbot gets the input from the user, it saves the input and the response which helps the chatbot with no initial knowledge to evolve using the collected responses. The program selects the closest matching response from the closest matching statement that matches the input. It then chooses the response from the known selection of statements for that response. This chatbot will have a textbox where we can give the user input, and the bot will generate a response for that statement. Our chatbot, Neobot uses rule-based approach i.e., it is trained according to rules. Based on this , it answers simple queries. The design of the Chatbot is to be taken place on the chatterbot framework. It becomes easier for the users to make chatbots using the Chatterbot library with more accurate responses. On top of this, the machine learning algorithms make it easier for the bot to improve on its own using the user’s input. Chatbots boost operational efficiency and bring cost savings to businesses while offering convenience and added services to internal employees and external customers. They allow companies to easily resolve many types of customer queries and issues while reducing the need for human interaction. With chatbots, a business can scale, personalize, and be proactive all at the same time—which is an important differentiator.

4.2) Functional modules design and analysis

Chatterbot makes it easy to create software that engages in conversation. Every time a chatbot gets the input from the user, it saves the input and the response which helps the chatbot with no initial knowledge to evolve using the collected responses.

With increased responses, the accuracy of the chatbot also increases. The program selects the closest matching response from the closest matching statement that matches the input, it then chooses the response from the known selection of statements for that response. We first need to run the command for installing the chatterbot library in python. Chatterbot comes with a data utility module that can be used to train the chatbots. Now there is training data for more than a dozen languages in this module. After we are done setting up the flask app, we need to add two more directories static and templates for HTML and CSS files. **index.html** file will have the template of the app and **style.css** will contain the style sheet with the CSS code. Execute the app.py code and check the terminal for an address shown. Click the address on the output, and you will get the app with the chatbot in the browser.

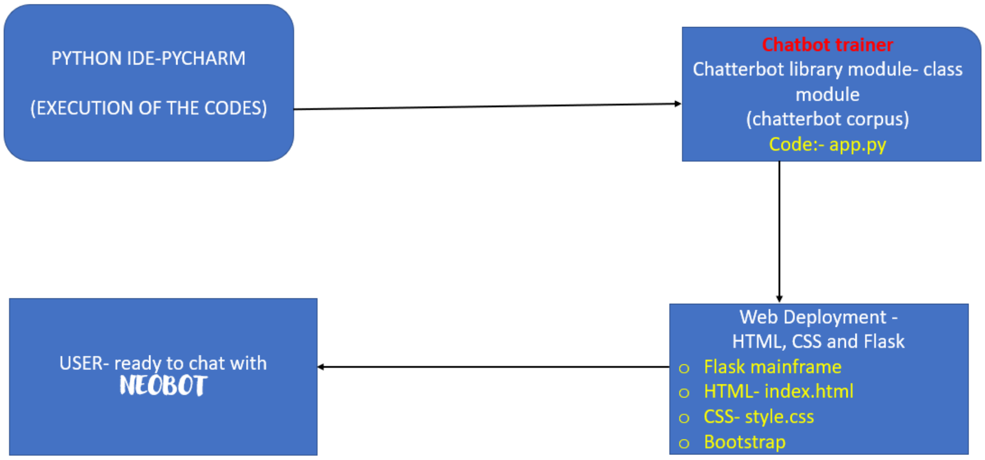


FIGURE-6

4.3) **Software Architectural designs**

The first step to designing the Chatbot is to know the scope and requirements like why chatbot, platform to launch chatbots and its limitations. The second step is to identify the inputs from users in the form of queries through text, voice, or images, from devices, and intelligence systems. The third step is to understand the User Interface (UI) elements, that we can see in our applications. UI elements are of five types they are: Command Line (CL), Graphical User Interface (GUI), Menu-Driven Interface (MDI), Form-Based Interface (FBI) and Natural Language Interface (NLI). After understanding user interface elements, the next step is to craft the first interaction and build a conversation. The final step of the Chatbot design process is testing, which is done on mobile and websites to know how it's working.

An architecture of Chatbot requires a candidate response generator and response selector to give the response to the user's queries through text, images, and voice.

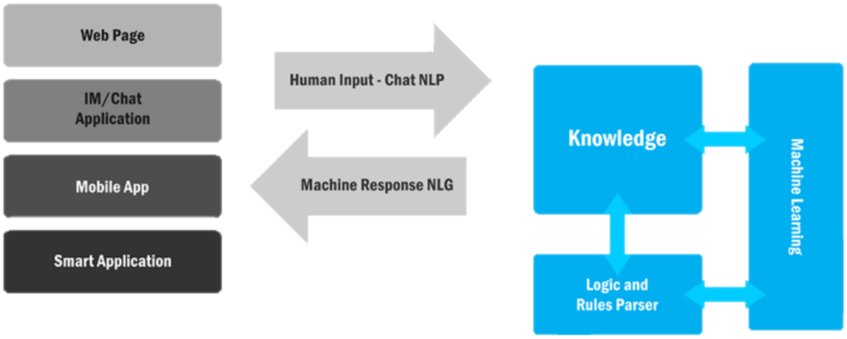
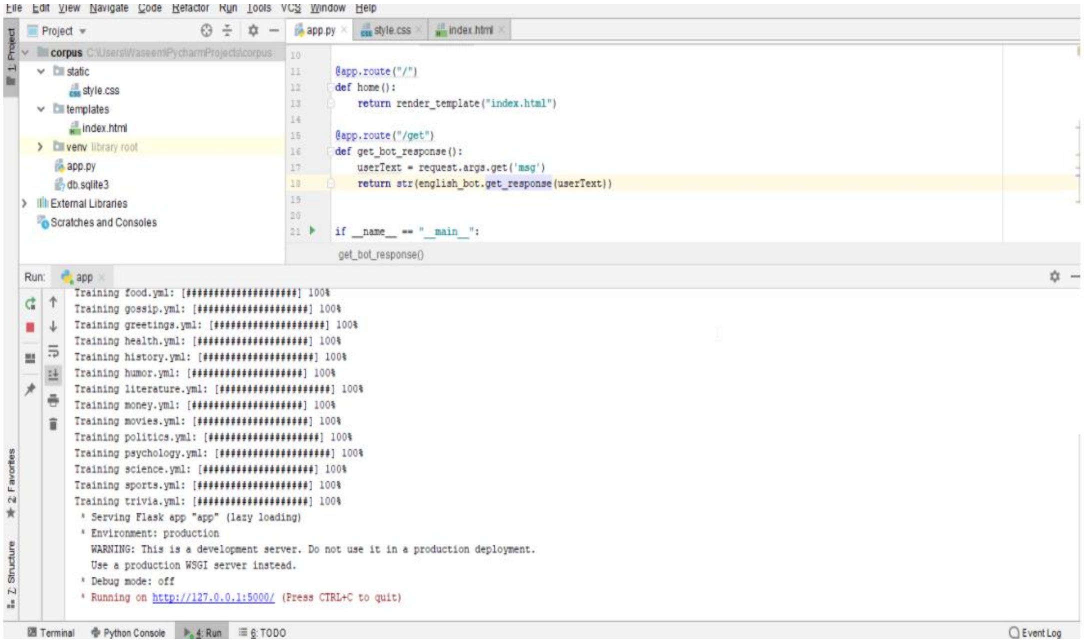


FIGURE-7

4.4) Subsystem services

For making the entire program, various subsystem functions were used, of which is the most important was the **app.py** as it contained the main program for the chatterbot deployment. **app.py** had imported the modules of the chatterbot library trainer. The use of **style.css** and **index.html** was not primary, in the chatbot website or platform. They designed the platform on which the chatbot would be deployed. Both directories were called in the **app.py** program and thus the program was executed as a whole, giving an address in its terminal, which on clicking, gave us the chatbot. We also have created another file, **about.html**, showing details of the creators and info of **Neobot**.



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

@app.route("/get")
def get_bot_response():
    userText = request.args.get('msg')
    return str(english_bot.get_response(userText))


if __name__ == "__main__":
```

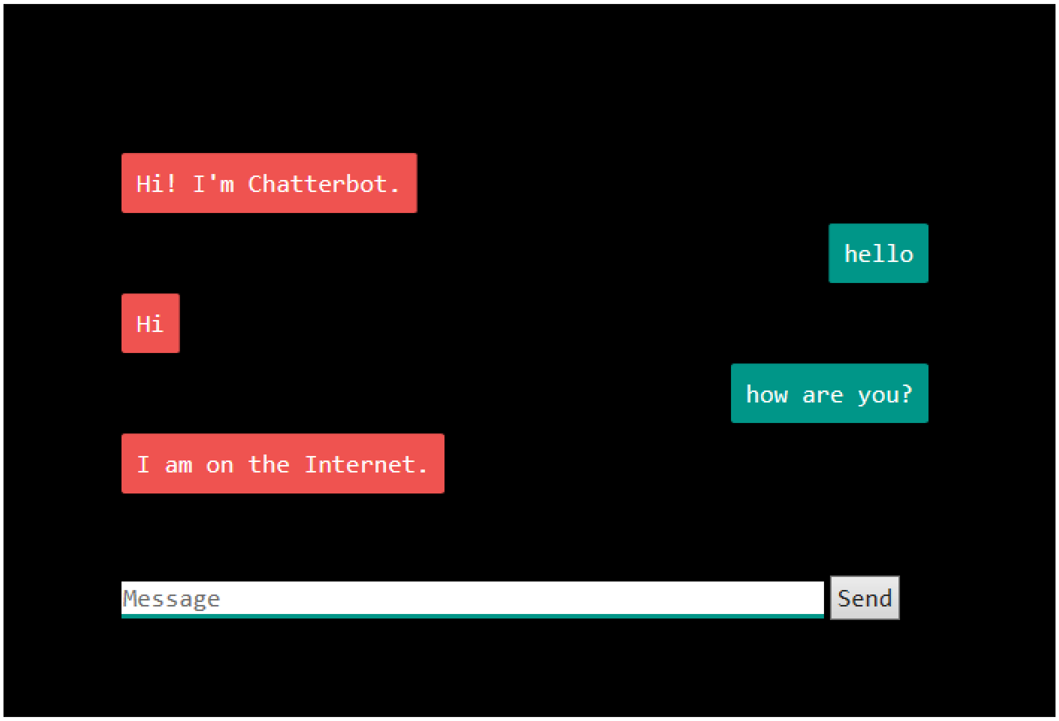
get_bot_response()

```
Training food.yml: [####################] 100%
Training gossip.yml: [####################] 100%
Training greetings.yml: [####################] 100%
Training health.yml: [####################] 100%
Training history.yml: [####################] 100%
Training humor.yml: [####################] 100%
Training literature.yml: [####################] 100%
Training money.yml: [####################] 100%
Training movies.yml: [####################] 100%
Training politics.yml: [####################] 100%
Training psychology.yml: [####################] 100%
Training science.yml: [####################] 100%
Training sports.yml: [####################] 100%
Training trivia.yml: [####################] 100%
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

FIGURE-8

4.5) User Interface designs

The program works on the pretext of an address shown in the terminal of the python program, which would take us to the chatbot website. The whole program works on GUI (graphic user interface). We must execute the code through the use on a touchpad or mouse on our laptop or computer for experiencing and using the chatbot. All the programs are a series of graphical and language elements that allow for human-computer interaction. However, Our Context and rule-based chatbot is designed to understand and respond to text inputs only.



Hi! I'm Chatterbot.

hello

Hi

how are you?

I am on the Internet.

FIGURE-9

4.6) Summary

Our chatbot, **Neobot** uses rule-based approach i.e., it is trained according to rules. Based on this , it answers simple queries. It becomes easier for the users to make chatbots using the Chatterbot library with more accurate responses. On top of this, the machine learning algorithms make it easier for the bot to improve on its own using the user’s input.

CHAPTER-5:
TECHNICAL IMPLEMENTATION & ANALYSIS

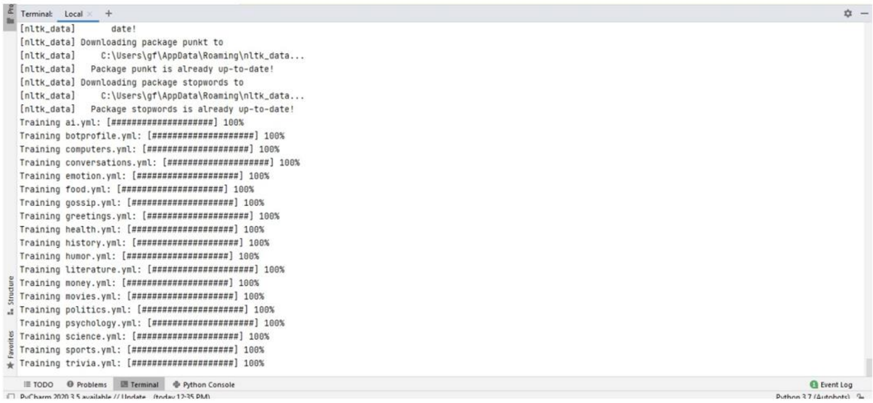
5.1) Outline

During the implementation of **Neobot**, I learned a great deal about chatbots and the effort it takes to create a “conversation-like” interface that can successfully support customers. An adoption strategy must first identify the required customer-facing knowledge to program the bot effectively. Based on my experiences, I have developed a chatbot implementation strategy that you should consider before you jump into the world of bots.

On developing an effective chatbot strategy, it is important to note that artificial intelligence is an exciting, emerging field that can help us not only serve customers in new ways, but also provide incredible insight into the customer experience. The implementation process of the chatbot is relatively easy, and the tools have well-documented implementation guides. While adoption is easy, it will be essential to establish a strategy that will allow the organization to experiment with this emerging channel, learn from the experience, and develop a more comprehensive approach over time.

5.2) Technical coding and code solutions

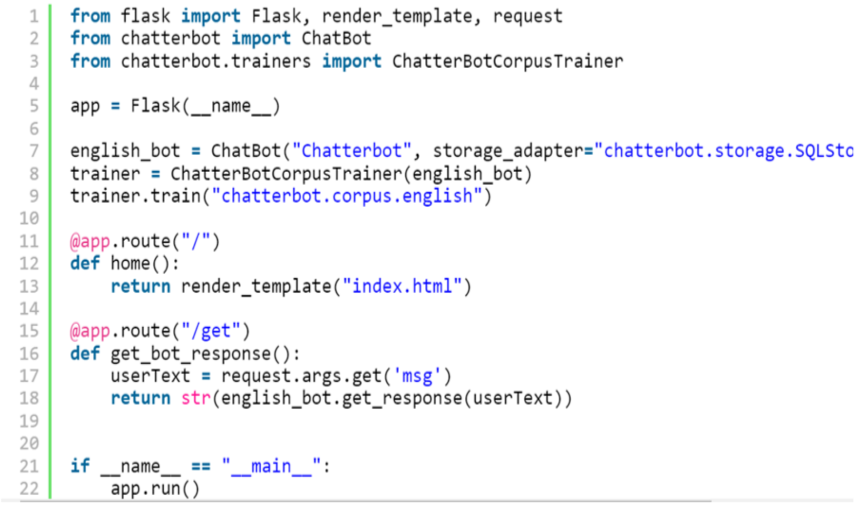
Trainer for our chatbot, Neobot



```
[nltk_data]       date!
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\gf\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\gf\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
Training ai.yml: [####################] 100%
Training botprofile.yml: [####################] 100%
Training computers.yml: [####################] 100%
Training conversations.yml: [####################] 100%
Training emotion.yml: [####################] 100%
Training food.yml: [####################] 100%
Training gossip.yml: [####################] 100%
Training greetings.yml: [####################] 100%
Training health.yml: [####################] 100%
Training history.yml: [####################] 100%
Training humor.yml: [####################] 100%
Training literature.yml: [####################] 100%
Training money.yml: [####################] 100%
Training movies.yml: [####################] 100%
Training politics.yml: [####################] 100%
Training psychology.yml: [####################] 100%
Training science.yml: [####################] 100%
Training sports.yml: [####################] 100%
Training trivia.yml: [####################] 100%
```

FIGURE-10

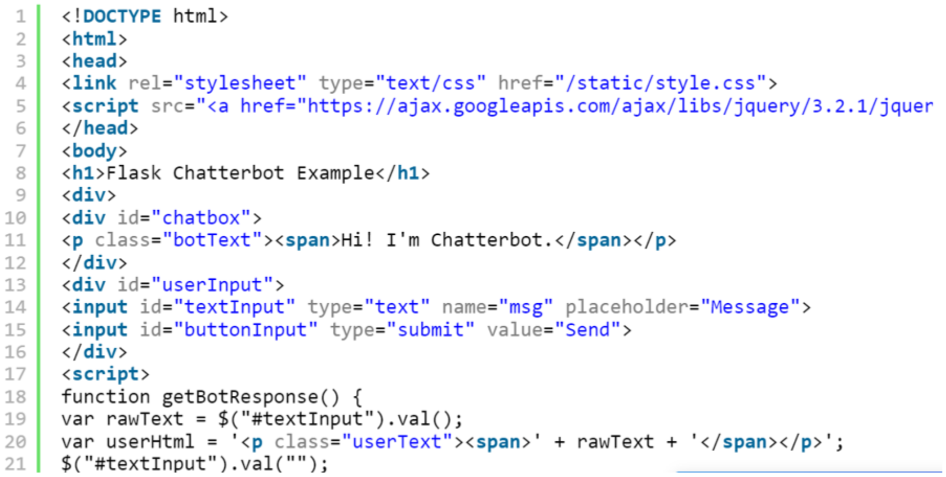
App.py = code for Neobot deployment



```python
from flask import Flask, render_template, request
from chatterbot import ChatBot
from chatterbot.trainers import ChatterBotCorpusTrainer

app = Flask(__name__)

english_bot = ChatBot("Chatterbot", storage_adapter="chatterbot.storage.SQLSto
trainer = ChatterBotCorpusTrainer(english_bot)
trainer.train("chatterbot.corpus.english")

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

@app.route("/get")
def get_bot_response():
    userText = request.args.get('msg')
    return str(english_bot.get_response(userText))


if __name__ == "__main__":
    app.run()
```

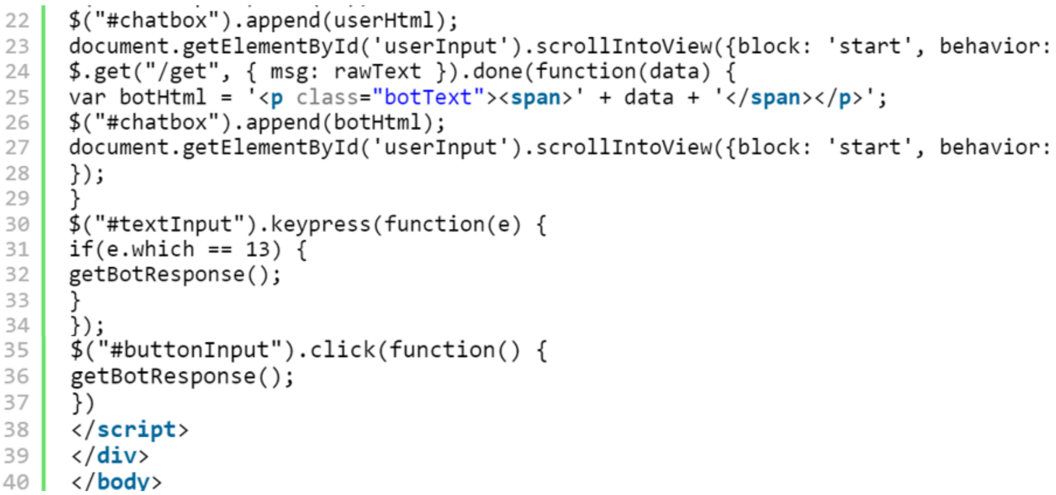
FIGURE-11

Index.html = code for **Neobot**



```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="/static/style.css">
<script src="<a href="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquer
</head>
<body>
<h1>Flask Chatterbot Example</h1>
<div>
<div id="chatbox">
<p class="botText"><span>Hi! I'm Chatterbot.</span></p>
</div>
<div id="userInput">
<input id="textInput" type="text" name="msg" placeholder="Message">
<input id="buttonInput" type="submit" value="Send">
</div>
<script>
function getBotResponse() {
var rawText = $("#textInput").val();
var userHtml = '<p class="userText"><span>' + rawText + '</span></p>';
$("#textInput").val("");
```

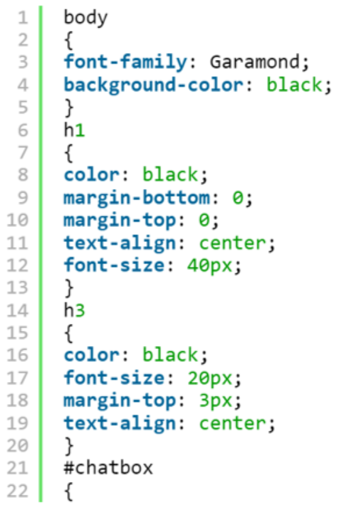
FIGURE-12



```
$("#chatbox").append(userHtml);
document.getElementById('userInput').scrollIntoView({block: 'start', behavior:
$.get("/get", { msg: rawText }).done(function(data) {
var botHtml = '<p class="botText"><span>' + data + '</span></p>';
$("#chatbox").append(botHtml);
document.getElementById('userInput').scrollIntoView({block: 'start', behavior:
});
}
$("#textInput").keypress(function(e) {
if(e.which == 13) {
getBotResponse();
}
});
$("#buttonInput").click(function() {
getBotResponse();
})
</script>
</div>
</body>
```

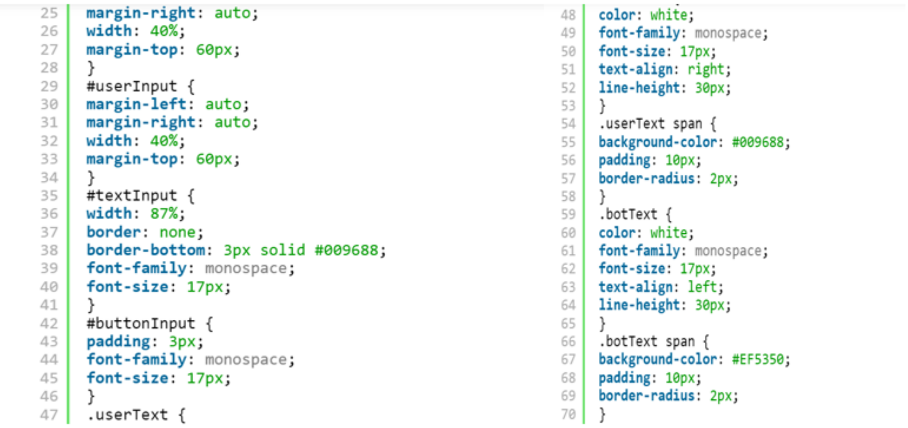
FIGURE-13

Style.css= style code for the CSS page for **Neobot**



```css
body
{
font-family: Garamond;
background-color: black;
}
h1
{
color: black;
margin-bottom: 0;
margin-top: 0;
text-align: center;
font-size: 40px;
}
h3
{
color: black;
font-size: 20px;
margin-top: 3px;
text-align: center;
}
#chatbox
{
```

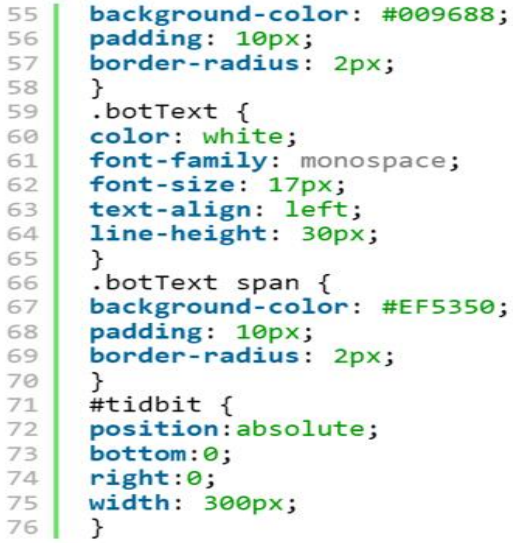
FIGURE-14



```css
margin-right: auto;
width: 40%;
margin-top: 60px;
}
#userInput {
margin-left: auto;
margin-right: auto;
width: 40%;
margin-top: 60px;
}
#textInput {
width: 87%;
border: none;
border-bottom: 3px solid #009688;
font-family: monospace;
font-size: 17px;
}
#buttonInput {
padding: 3px;
font-family: monospace;
font-size: 17px;
}
.userText {
```

```css
color: white;
font-family: monospace;
font-size: 17px;
text-align: right;
line-height: 30px;
}
.userText span {
background-color: #009688;
padding: 10px;
border-radius: 2px;
}
.botText {
color: white;
font-family: monospace;
font-size: 17px;
text-align: left;
line-height: 30px;
}
.botText span {
background-color: #EF5350;
padding: 10px;
border-radius: 2px;
}
```

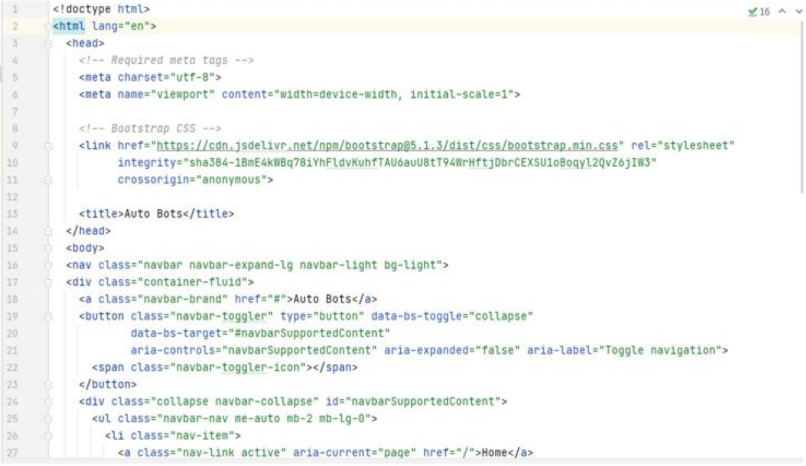
FIGURE-15



```css
background-color: #009688;
padding: 10px;
border-radius: 2px;
}
.botText {
color: white;
font-family: monospace;
font-size: 17px;
text-align: left;
line-height: 30px;
}
.botText span {
background-color: #EF5350;
padding: 10px;
border-radius: 2px;
}
#tidbit {
position:absolute;
bottom:0;
right:0;
width: 300px;
}
```

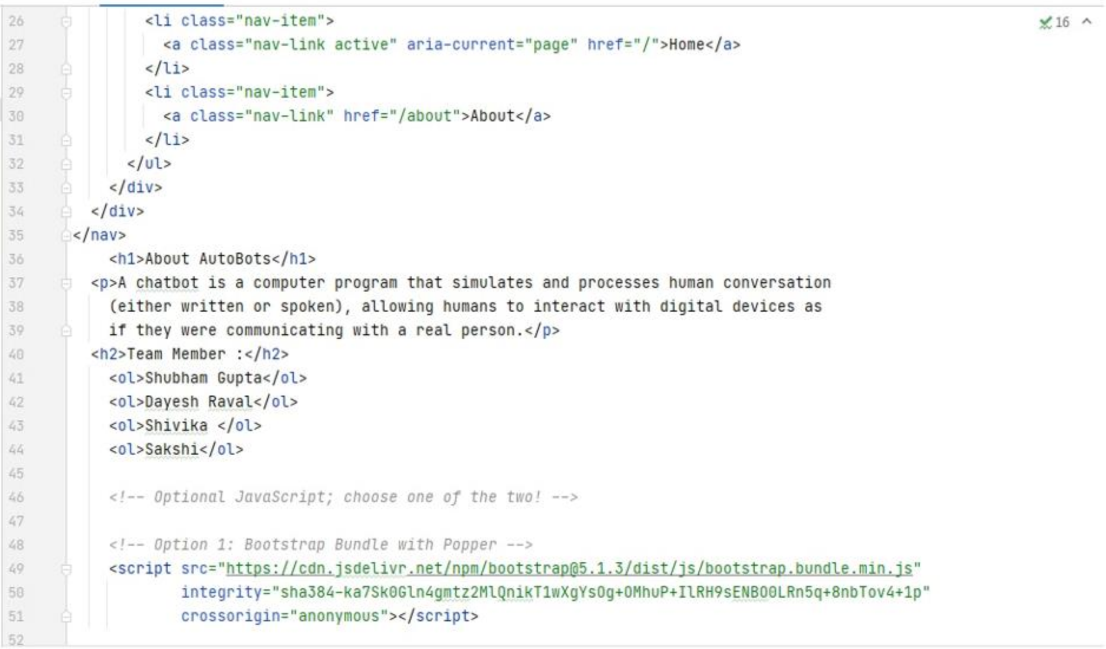
FIGURE-16

About.html= Neobot info and the team members



```html
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
          integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
          crossorigin="anonymous">

    <title>Auto Bots</title>
  </head>
  <body>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Auto Bots</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
            data-bs-target="#navbarSupportedContent"
            aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="/">Home</a>
```

FIGURE-17



```
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="/">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/about">About</a>
        </li>
      </ul>
    </div>
  </div>
</nav>
    <h1>About AutoBots</h1>
  <p>A chatbot is a computer program that simulates and processes human conversation
    (either written or spoken), allowing humans to interact with digital devices as
    if they were communicating with a real person.</p>
  <h2>Team Member :</h2>
    <ol>Shubham Gupta</ol>
    <ol>Dayesh Raval</ol>
    <ol>Shivika </ol>
    <ol>Sakshi</ol>

    <!-- Optional JavaScript; choose one of the two! -->

    <!-- Option 1: Bootstrap Bundle with Popper -->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
            integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
            crossorigin="anonymous"></script>

```

FIGURE-18

5.3) <u>Working layout of the forms</u>

No forms have been used in making of the chatbot. The chatbot can be executed by using the address in the terminal of the python IDE to get the desired output.

5.4) **Prototype submission**

A prototype of our chatbot model was used to experiment and create analysis for its performance review. The prototype was delivered to a small group of people, in different backgrounds, through the address generated in the terminal of the python IDE, **Pycharm**. Success and thus accuracy of the chatbot was calculated on basis on the question-response parameter efficiency.

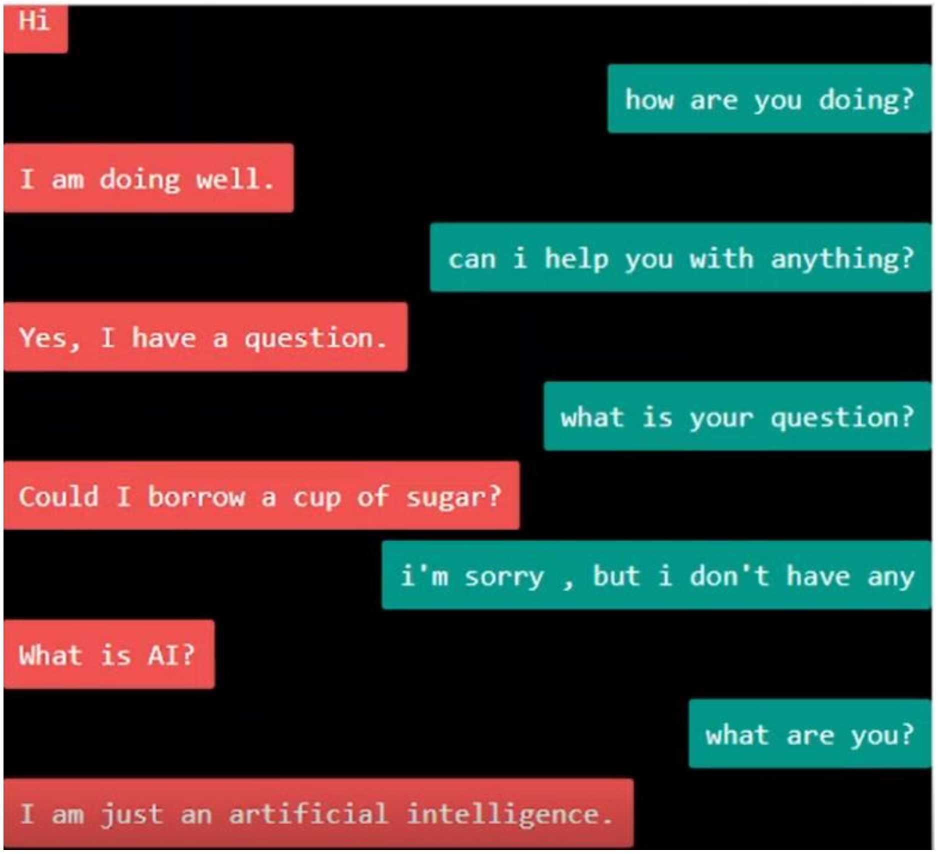
5.5) Test and validation

This is the list of the sample tests for our chatbot:-

STUDENT NAME	**PLATFORM FOR EXECUTION**
DAYESH RAVAL	ONE PLUS NORD PHONE
SHUBHAM GUPTA	ASUS LAPTOP
SHIVIKA BANSAL	SAMSUNG TABLET
SAKSHI JAISWAL	NOKIA 3310

TABLE-2

On testing on one of the devices, we got the following output:-



Hi

how are you doing?

I am doing well.

can i help you with anything?

Yes, I have a question.

what is your question?

Could I borrow a cup of sugar?

i'm sorry , but i don't have any

What is AI?

what are you?

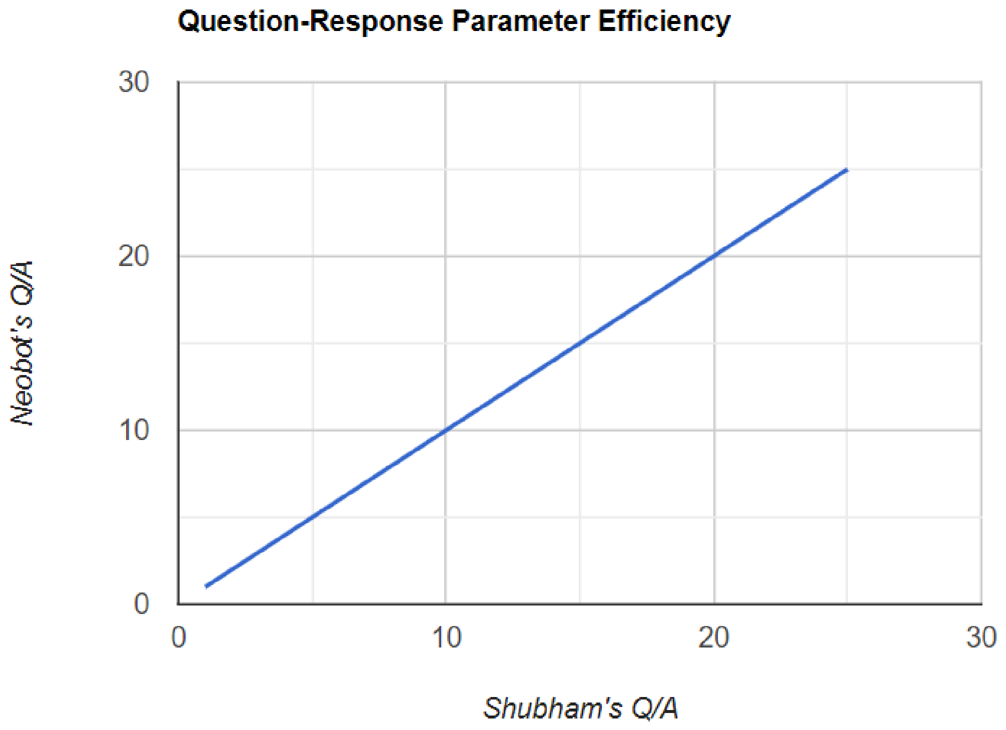
I am just an artificial intelligence.

FIGURE-19

This shows us that among all the tests conducted on the chatbot, the one execute on the laptop of **Shubham** showed us the above and correct output and gave a high question-response parameter efficiency. For rest, output was not correct per question. **Sakshi** could not execute the chatbot, while Shivika and Dayesh got low question-respone parameter efficiency.

5.6) **<u>Performance Analysis</u>**

As the best question-response parameter efficiency was for **Shubham**, his performance anlysis graph is as follows:-



GRAPH-1

5.7) **Summary**

Chatbots are already transforming industries across the board—and they're only going to become more impactful as technology evolves. In this activity, we will be using a Python library, chatterbot, to implement a chatbot. This library generates automated responses to the user input. The responses are based on machine learning algorithms implemented in the library. Our goal is to make sure that the chatbot is trained in multiple languages to cater to different communities. These features make it easier for the chatbot to engage in conversations through different mobile apps and websites. It saves input data from the user and evolves over time. The accuracy of the chatbot response increases over time as it learns from the user.

CHAPTER-6:

PROJECT OUTCOME AND APPLICABILITY

6.1) Outline

As we begin down the path of creating a chat-based & utilitarian mobile experience for museums, it’s important to note that we on the Studio team do not consider ourselves to be experts in chatbots, artificial intelligence or conversational interfaces. Acknowledging this, our primary goal right now is to learn as much as possible (as quickly as possible) from the experts that do exist within the bot domain.

6.2) Key implementations outline of the system

This flask chatterbot, **Neobot** is inclined towards performing a specific task for the user. Normally, any chatbot is merely a computer program that fundamentally simulates human conversations. It can answer questions formulated to it in natural language and respond like a real person. With the increasing advancements, there also comes a point where it becomes difficult to work with chatbots. Since true artificial intelligence is still out of reach, it becomes difficult for any chatbot to completely fathom the conversational boundaries when it comes to conversing with a human. Not being able to respond correctly in addition to poor comprehension skills, has been the most frequent errors for the chatbots, adding a personality to a chatbot is still a benchmark that seems within our reaches. **NeoBot**, our chatterbot makes it easy to create software that engages in conversation. The users can start a conversation where they wish to begin with, but it is not feasible to immediately respond as the logs of chats are stored and referred to in the future.

Let's look for tickets!

Ok

From New York
To Seatle
Departing on Sept 15
Returning on Sept 19
for 2 Adults

Is this info correct?

Yes

I have found 17 results

FIGURE-20

6.3) Significant project outcomes

When testing the last prototype, we got findings suggesting that the tests did not have a problem with getting information from a chatbot instead of a human. The information that they got was not seen as less trustworthy, this could be supported by the fact that the chatbot provided a source for the information it gave. It has been interesting to investigate how the participants interacted with the chatbot and how they reported on it afterwards. Still, we have to stress the fact that the chatbot was not very intelligent and that the evaluators had to adjust their language to match the chatbots.

6.4) Project applicability on Real-world applications

We are entering the age of automation, and the rise of artificial intelligence is already evident in the form of chatbots. What's common between the customer service offered by a hotel, a retailer, and a bank these days? Chatbots! Chatbots today are being deployed across industries to assist customers (customer service), or engage with customers (sales and marketing), or do both. Bots can automatically and intelligently route tickets to the right agents. These bots can also detect the intent in customer conversations and help teams overcome common challenges such as preventing 'thank-you' responses from being reopened. You can embed a feedback form in your chatbot widget and trigger a popup when a customer is about to leave your website. By doing this, you can identify the reasons for drop-offs and make improvements accordingly.

6.5) Inference

Our AI-powered interface, Neobot will stimulate human conversation and engage with customers and provide the best user satisfaction across all fields.

CHAPTER-7:

CONCLUSIONS AND RECOMMENDATION

7.1) Outline

After months of reviewing our project, assessing it, modifying our goals in general, we have concluded that our project is enabling human- machine communication at the best level. Our team, **AUTOBOTS** have collaboratively made this project for empowering human-machine interface for its smooth application and greater benefits.

7.2) Limitations/Constraints of the system

Though, even after months of coding and regressive attempts to be error-free, we do have certain limitations on our project. The database has not been made for a perfect communication with the machine, showcasing limited domain knowledge. We can certainly work on the template of our chatbot for different types of functionalities. Lastly, we could have incorporated the voice recognition feature to our chatbot.

7.3) Future Enhancements

As our project has progressed in this past time, we are very much engaged & excited for various future endeavors and enhancements regarding our project. We plan to focus on various updates and integrations in our chatbot, implementing voice commands, recognition along with natural language processing (NLP). To fully achieve our final goal of a more technical growth, we plan to launch this app for all the users worldwide so as increase reach and technological experience of our users.

7.4) Inferences

At last, we infer that there has been immense research to find various methodologies to bring out the general mass in streamline with the upcoming technological development and ventures. As junior developers, we have the power to develop better technologies to make the world a better and stress-free place.

References

https://www.edureka.co/blog/how-to-make-a-chatbot-in-python/

cobusgreyling.medium.com

Bayan Abu Shawar and Eric Atwell, 2007 “Chatbots: Are they Really Useful?”

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 06 Issue: 11 | Nov 2019 www.irjet.net p-ISSN: 2395-0072 © 2019, IRJET | Impact Factor value: 7.34 | ISO 9001:2008 Certified Journal | Page 391 CHATBOT IN PYTHON Akshay Kumar1, Pankaj Kumar Meena2, Debiprasanna Panda3, Ms. Sangeetha4