BUILDING A CHATBOT USING PYTHON AND FLASK

INTRODUCTION

- Developing a chatbot using Flask, a Python web framework, involves creating a web application that can receive and respond to user messages.

Prerequisites

- Basic understanding of python programming.
- Familiarity with web development and API's.

Here's a high-level overview of the steps to develop a basic chatbot using Flask:

- Set up the Environment

- Create the Flask app

- Import Dependencies

- Create Routes

- Process User Messages

- Create Templates

- Handle User Input

- Generate Bot Responses

- Display Conversations

- Run the Flask app

Sample Code:

```python
from flask import Flask, request, jsonify

 app = Flask(__name)

@app.route('/')

def chatbot_page():

        return render_template('chatbot.html')

@app.route('/get_response', methods=['POST'])

def get_response():

user_message = request.form['user_message']
```

```
    app.run(debug=True)```# Implement your chatbot logic here to generate a
response based on the user's message.

        Bot_response = "This is a placeholder response."

        return jsonify({'bot_response': bot_response})

    if __name__ == '__main__':
```

1. Set up the Environment:

Create a new directory for your project and navigate to it using the terminal.

Install Python in your system. Visit the link
https://www.python.org to download the latest release of Python

After Python installation, execute the below command to install virtualenv package inside the terminal:

```
pip install virtualenv
```

2. Install Required Libraries:

Install Flask:

```
pip install Flask
```

Install OpenAi Library:

```
pip install openai
```

3. Create Flask App Structure:

- Create a new Python file, e.g., app.py, to hold your Flask application.

Import necessary modules:

From flask import Flask,render_template

Initialize the Flask app:

```
#app.py
#import files
from flask import Flask, render_template,
request
app = Flask(__name__)
@app.route("/")
def home():
    return "Hello, This is Flask Application"
if __name__ == "__main__":
    app.run()
```

4. Create HTML Template:

- Create a folder static in your project directory, place the image that needs to be displayed in the UI and name it as chatbot1.png

- Create a templates folder in your project directory.

- Inside the templates folder, create an HTML file, e.g., index.html. This file will be the frontend of your chatbot.

```html
<!DOCTYPE html>
<html>
<head>
    <title>GenAI-Bot</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
<script src=https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js></script>
    <style>
* {
        box-sizing: border-box
```

```css
        }

        /* Set height of body and the document to 100% */

            body, html {

                height: 100%;

                margin: 0;

                font-family: Arial;

            #chatbox {

            margin-left: auto;

            margin-right: auto;

            width: 40%;

            margin-top: 60px;

        }#userInput {

            margin-left: auto;
```

```css
            Margin-right: auto;

                width: 40%;

                margin-top: 60px;

            }

            #textInput {

                width: 90%;

                border: none;

                border-bottom: 3px solid black;

                font-family: monospace;

                font-size: 17px;

            }.userText {

                color: white;

                font-family: monospace;

                font-size: 17px;
```

```css
Text-align: right;

    line-height: 30px;

}

.userText span {

  background-color: #808080;

  padding: 10px;

  border-radius: 2px;

}

.botText {

  color: white;

  font-family: monospace;

  font-size: 17px;

  text-align:left;

  line-height: 30px;
```

```css
.botText span {

    background-color: #4169e1;

    padding: 10px;

    border-radius: 2px;}b

#tidbit {

  position: absolute;

  bottom: 0;

  right: 0;

  width: 300px;}

.boxed {

  margin-left: auto;

  margin-right: auto;

  width: 78%;

  margin-top: 60px;

  border: 1px solid green;

}
```
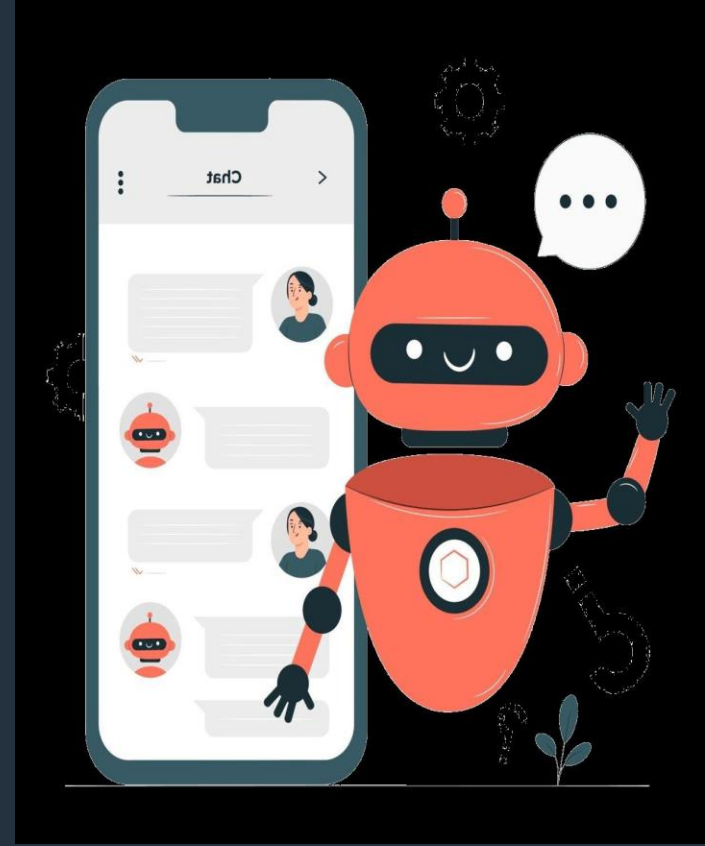
Goals:

- Useful in communication source.

- Acquire leads from command.

- To grow our customer service in business.

- Responding algorithm is the main goal.

- Developing a chatbot for social media platforms to engage with users and answer their questions.

- Building a chatbot for language learning or practicing conversational skills .

Conclusion:

Chatbot is one of the simple ways to transport data from a computer without having to think for proper keywords to look up in a search or browse several web pages to collect information

Team:

P. AMANDA JUDITH
S. AARATHI NISHA
B. BRIALLEN REBA
F. ANTONY PACKIA SWEETY
E.BEGENCY