CREATE A CHATBOT IN PYTHON

TEAM MEMBERS

1. SUYAMBULAKSHMI V
2. SHARINITHA R
3. AYESHASITHIKA S
4. PIRITHIBA M

Phase 2 – Document submission

OBJECTIVE:

- The Objective of this project is to create a high-quality support to users, ensuring a positive user experience and customer satisfaction chatbot in Python that provides exceptional customer service, answering user queries on a website or application.

PROGRAM:

- This is the sample program,

```
import json
import string
import random
import nltk
import numpy as num
from nltk.stem import WordNetLemmatizer # It has the ability to lemmatize.
import tensorflow as tensorF # A multidimensional array of elements is
represented by this symbol.
from tensorflow.keras import Sequential # Sequential groups a linear stack of
layers into a tf.keras.Model
from tensorflow.keras.layers import Dense, Dropout

nltk.download("punkt")# required package for tokenization
nltk.download("wordnet")# word database
import re
import long_responses as long

def message_probability(user_message, recognised_words,
single_response=False, required_words=[]):
```

```
  message_certainty = 0
  has_required_words = True


  #counts how many words are present in each predefined message
  for word in user_message:
    if word in recognised_words:
      message_certainty +=1


  #calculates the percent of recognised words in a user message
  percentage = float (message_certainty) / float(len(recognised_words))

  #checks that the required words are in the string
  for word in required_words:
    if word not in user_message:
      has_required_words = False
      break

  if has_required_words or single_response:
    return int(percentage+100)
  else:
    return 0

def check_all_messages(message):
  highest_prob_list = {}


  def response(bot_response, list_of_words, single_response=False,
required_words=[]):
    nonlocal highest_prob_list
    highest_prob_list[bot_response] = message_probability(message,
list_of_words, single_response, required_words)


    #response-------------------------------------
    response("Hello!",["hello","hi","sup","hey","heyo","whatsup"],
single_response=True)
    response('I\'m doing fine, and You?',["how","can","i","help","you"],
required_words=['how'])
    response("Thank You!",['i','love','doing','online','courses'],
required_words=['online','palace'])

    response(long.R_EATING, ['what','you','eat'],
required_words=['you','eat'])


    best_match = max(highest_prob_list, key=highest_prob_list.get)
    #print(highest_prob_list)
```

```
    return long.unknown() if highest_prob_list[best_match] < 1 else
best_match



def get_response(user_input):
  split_message = re.split(r'\s+|[,;?!.-]\s*', user_input.lower())
  response = check_all_messages(split_message)
  return response

while True:
  print('Bot: ' + get_response(input('You: ')))
```

- Here mentioned program is called in the above program.

```
import random

R_EATING = "I don't like eating anything because I'm a bot obviously!"


def unknown():
    response = ['Could you please re-phase that?',
                "...",
                "sounds about right",
                "what does that mean?"][random.randrange(4)]
    return response
```

OUTPUT:

```
You: what do you like to eat?
Bot: I don't like eating anything because I'm a bot obviously!
You: what is the weather like today?
Bot: What does that mean?
You:
```

- **STEP 1:** We have developed a website in order to implement the chatbot. This is our website link - https://technerdscentre.neocities.org/chatbot/ . This website is

created by using the basic knowledge of HTML, CSS, JavaScript. and it is published with the help of neocities website.

- **STEP 2:** By learning some important features of **Machine Learning,** For Example Data wrangling, neural networks, natural language processing,etc. And also some of the basic libraries we have installed the **Visual Studio Code** , They are, JSON,string, random, nltk, pytorch, flask app, request.
- **STEP 3:** We are using Visual Studio code plateform to implement this code
- **STEP 4:** With the help of flask app and javascript , we are able to connect the python chatbot in our website.
- **STEP 5:** Here we will test our project