# HEY YAAR!!

# A COUNSELLOR SOFTWARE WITH ML

**A report of final year project(2022-2023)**


**Submitted by**

**GROUP-5**

**HIMANSHI MISHRA**

**HARSHITA SHUKLA**

**JUNAID MOHSIN**

**Supervised by**

Dr. Abhinav Juneja


**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

**KIET GROUP OF INSTITUTIONS, GHAZIABAD, UTTAR PRADESH(Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow, UP, India**


**Session 2019-2023**

# Introduction:-

Everybody has stress and/or mental health related issues. In fact, according to a 2015 study (globally) the number of people who suffered from some form of depressive disorder worldwide was estimated to be over 322.48 million people. And according to another 2017 study, more than 14 percent of the total populace in India suffers from variations of mental disorders; thereby, constituting a major cause of distress in people's life with impact on the well-being of the society and the social quality thereof. Unchecked stress can also lead to number of health issues, effecting one physiologically and taking a toll on your body and one's daily life. Indeed, roughly 50-80% of all physical illnesses are caused by stress where the effect thereof is believed to be the main cause of these dysfunctions and is correlated with increase in risk diabetes, cardiovascular (heart) diseases, sexual malfunction, et cetera, and other physical ailments such as migraines, skin disorders, epilepsy; whereof each of these illnesses – and many others – are psychosomatic in nature (prompted or exacerbated by mental conditions such as stress).

# Objectives:-

The main objective of this study is to bridge the gap between mental health management systems and the users by machine learning via an interactive chatbot that performs clinical analysis and predicts the mental health problems by searching reliable databases and thereby handles the stress of people. A user-friendly chatbot utilizing natural language processing technique will simulate the role of a psychologist, counselor, or stress specialist who provides virtual counseling. The successful implementation of this project is expected to provide people with access to free treatment to overcome their plights and predicaments, and reduce the time and human effort required to determine the best recommendations and solutions for stress management.

# Project Overview:-

- **DATASET COLLECTION AND ANALYSIS**

We have collected the dataset from people by sending google form.

In google form , we have made different columns like topic, title and id of problem. By using this information we made a database.

- **ALGORITHM**

As we are dealing with real world problem and collecting data in terms of natural language. So we have used natural language processing algorithm.

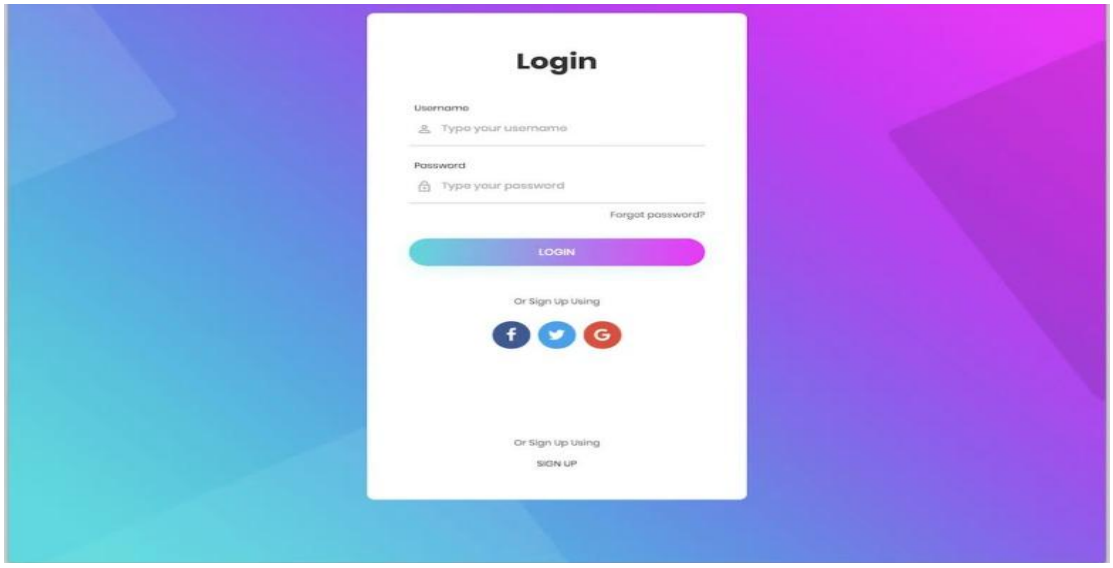
- **WEBSITE**

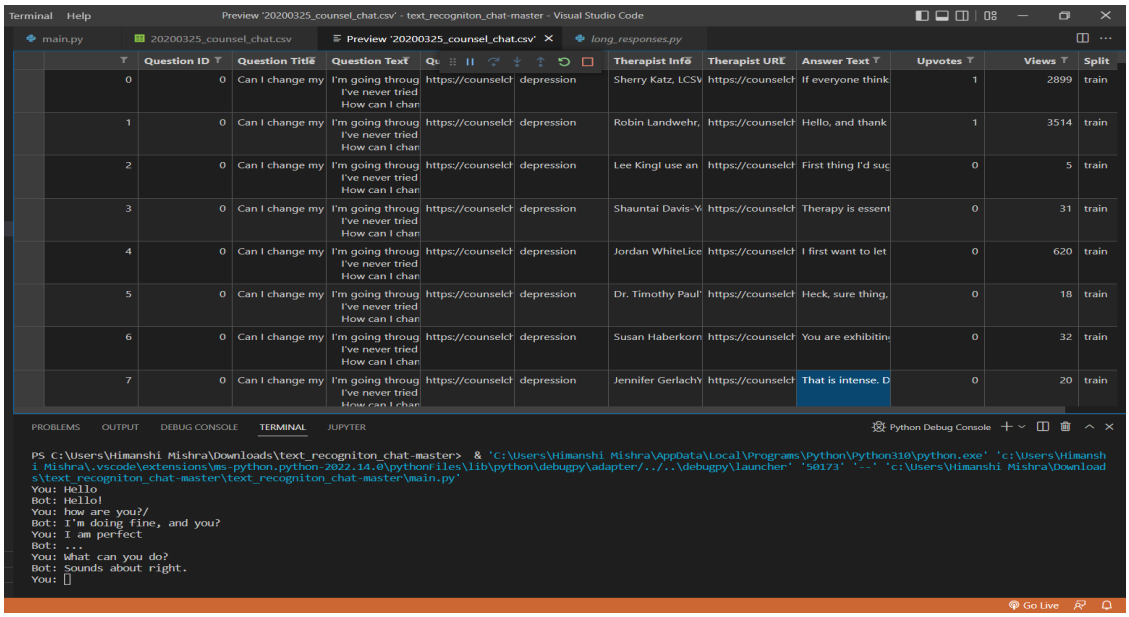
We have made one website whose name is Hey counsellor. So here, we have made login page, where the user login and then by searching their problem get the appropriate solution.

# Implementation:-

Our project works in following ways:-

Firstly user do a login by entering his name.
Then user will enter his problem .
According to the problem appropriate solution will be provided.
Problem can be anything whether family issues, studies problem or even addiction.



```python
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

    # Must either have the required words, or be a single response
    if has_required_words or single_response:
        return int(percentage * 100)
    else:
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER — Python Debug Console

```
PS C:\Users\Himanshi Mishra\Downloads\text_recogniton_chat-master>  & 'C:\Users\Himanshi Mishra\AppData\Local\Programs\Python\Python310\python.exe' 'c:\Users\Himanshi Mishra\.vscode\extensions\ms-python.python-2022.14.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '50173' '--' 'c:\Users\Himanshi Mishra\Downloads\text_recogniton_chat-master\text_recogniton_chat-master\main.py'
You: Hello
Bot: Hello!
You: how are you?/
Bot: I'm doing fine, and you?
You: I am perfect
Bot: ...
You: What can you do?
Bot: Sounds about right.
You:
```

Ln 59, Col 49 Spaces: 4 UTF-8 LF Python 3.10.5 64-bit Go Live Prettier

```python
import random

R_EATING = "I don't like eating anything because I'm a bot obviously!"
R_ADVICE = "if I were you, I would go to the internet and type exactly what you wrote there!"


def unknown():
    response = ["Could you please re-phrase that? ",
                "...",
                "Sounds about right.",
                "What does that mean?"][
        random.randrange(4)]
    return response

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER — Python Debug Console

```
PS C:\Users\Himanshi Mishra\Downloads\text_recogniton_chat-master>  & 'C:\Users\Himanshi Mishra\AppData\Local\Programs\Python\Python310\python.exe' 'c:\Users\Himanshi Mishra\.vscode\extensions\ms-python.python-2022.14.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '50173' '--' 'c:\Users\Himanshi Mishra\Downloads\text_recogniton_chat-master\text_recogniton_chat-master\main.py'
You: Hello
Bot: Hello!
You: how are you?/
Bot: I'm doing fine, and you?
You: I am perfect
Bot: ...
You: What can you do?
Bot: Sounds about right.
You:
```

# Conclusion:-

In this project, we build customer support chatbot that helps companies to have 24 hours of automated responses. After analyzing the dataset and understanding the importance to have automated responses to customers and companies, we start exploring existing techniques used for generating responses in the customer service field. Then, we attempt to try natural language processing.

FUTURE WORK:-

In future work, we plan to incorporate other similarity measures such as soft cosine similarity. Also, we plan to improve the experiments by increase the vocabulary size and try to increase the epoch parameters to reach 100 after providing proper infrastructure. We further can add more data for the training by taking benefits from the queries without responses and translate non-English queries.