MOOD ANALYSIS

Project submitted to:
SRM University-AP, Andhra Pradesh

for the partial fulfilment of the requirements to award the degree of

Bachelor of Technology
In
Computer Science Engineering
School of Engineering and Sciences

Submitted by:
Jyothi Samhitha Katragadda (AP22110010928)
Pardhiv Thallamsetti(AP22110010936)
Navya Sree Pinnamaneni(AP22110010981)
Priyanka Kumari Shah(AP22110011520)

Under the guidance of:
Mr. Repaka Subha Sandeep
Ms. Karnena Kavitha Rani

SRM University - AP
Neerukonda, Mangalagiri, Guntur,
Andhra Pradesh – 522 240

[December, 2023]

CERTIFICATE

Date: 06.12.2023

This is to certify that the work present in this project titled "**Mood Analyzer**" has been carried out by the group comprising of **Jyothi Samhitha Katragadda, Pardhiv Thallamsetti, Navya Sree Pinnamaneni and Priyanka Kumari Shah** under our supervision. The work is genuine, original, and suitable for submission to SRM University AP for the award of **Bachelor of Technology** in the **School of Engineering and Sciences**.

Supervisors

Mr. Subha Sandeep Repaka

Ms. Karnena Kavitha Rani

SRM University, Andhra Pradesh

ACKNOWLEDGEMENT

We extend our heartfelt gratitude to our esteemed mentors, Mr. Repaka Subha Sandeep and Ms. Karnena Kavitha Rani, for entrusting us with the opportunity to embark on this project. This undertaking has served as an invaluable learning experience, providing profound insights into various core concepts of object-oriented programming, including data encapsulation, exception handling, objects, classes, constructors and more.

Our journey through this project has been enriched by the unwavering support and guidance extended to us by our mentors. Their expertise, encouragement, and insightful suggestions have been instrumental in steering us through the intricacies of the project, fostering our understanding of complex programming concepts, and enhancing our problem-solving skills.

We express our deep appreciation for the dedication and mentorship demonstrated by Mr. Repaka Subha Sandeep and Ms. Karnena Kavitha Rani throughout the course of this endeavour. Their commitment to our learning and professional growth has left an indelible mark on our academic and practical journey.

Once again, thank you for your invaluable mentorship, which has played a pivotal role in shaping our understanding and competence in the realm of object-oriented programming. We are sincerely grateful for the opportunity and the knowledge imparted, and we look forward to carrying forth the lessons learned into our future endeavours.

INTRODUCTION

Problem Statement:

You are tasked with developing a simple interactive program that collects user information, assesses their mood on a scale of 0.5 to 5.0, and provides appropriate responses based on their mood level. The program should perform the following tasks:

1. User Information:

- Use a class “User” to prompt the user for their name and display a welcoming message.

- Ask for the user's gender.

2. Input Validation:

- Validate the user's age input to ensure it is a positive integer.

- Validate the user's mood input to ensure it falls within the specified range (0.5 to 5.0).

3. Display Mood Scale:

- Display a mood scale ranging from 0.5 to 5.0, providing descriptions for each level.

4. Write to File:

- Write the collected user information (name, gender, age, mood) to "user_responses.txt."

5. Mood Assessment:

- Based on the user's provided mood level, provide appropriate responses:

  - If mood is >= 3.5, display a positive message and offer stress relief game suggestions.

  - If mood is >= 3.0, display an encouraging message and offer stress relief game suggestions.

  - If mood is >= 1.5, acknowledge the user may be feeling low and inquire about consistency. Offer helpline information if the low mood is consistent over time.

  - If mood is >= 0.5, express understanding and inquire about consistency. Offer helpline information if the low mood is consistent.

6. Additional Features:

- The program should be designed with input validation to handle unexpected user inputs gracefully.

- Implement functions for modular and organized code.

- Ensure error handling for file operations (opening "user_responses.txt").

7. Output Format:

- The program should have clear and concise output messages.

- Ensure the program's output is user-friendly and encourages positive interaction.

Your goal is to create a well-structured and efficient program that fulfills these requirements, providing a positive and supportive experience for the user.

Aim of the Project:

The primary objective of the project is to create an interactive and empathetic mental health and well-being program. The key objectives include:

1. User Information Collection:
Gather essential user information such as name, gender, age, and mood.

2. Input Validation:
Validate user inputs to ensure accuracy and gracefully handle unexpected inputs.

3. Mood Assessment:
Utilize a mood scale to quantify the user's emotional state on a scale from 0.5 to 5.

4. Data Recording:
Write user responses to a file ("user_responses.txt") for future reference.

5. Personalized Support:
Assess the user's mood level and provide personalized supportive messages and suggestions based on their emotional state.

6. Resource Provision:
Offer additional resources, including helpline numbers and stress relief game suggestions, to assist users in managing their well-being.

7. Positive Closure:
Conclude the interaction with positive messages, encouraging self-love and self-care, and wishing the user the best.

ALGORITHM

1. Include Necessary Headers:

Include C++ standard headers: iostream, fstream, string, and limits.

2. Initialize Variables:

Declare variables userName, gender, age, and mood of appropriate data types.

3. Define User Class:

a. The class User is defined with private data members for the user's name, gender, age, and mood.

b. The constructor initializes age and mood to default values.

4. User Information Input:

a. Define function getUserInfo

b. Output prompt for the user's name.

c. Use getline to read the user's name and display a welcome message.

d. Use a loop to:

- Prompt the user for their gender and store it in the variable gender.

- Convert the gender input to lowercase for case-insensitive validation.

- Check if the input is valid (either "male," "female," or "other"). If not, repeat the loop.

e. Use a loop to:

- Prompt the user for their age and store it in the variable age.

- Validate the age input (between 10 and 100). If not valid, repeat the loop.

f. Use a loop to:

- Prompt the user to rate their mood on a scale of 0.5 to 5.0 and store it in the variable mood.

- Validate the mood input. If not valid, repeat the loop.

5. Write User Information to File:

a. Define function writeToFile

b. Open an output file stream (ofstream) in append mode.

c. Write user information to the file in a formatted manner.

d. Close the file.

6. Main Function:

a. Declare variables: userName, gender, age, and mood.

b. Call getUserInfo with references to these variables to collect user information.

c. Call writeToFile with the collected user information to store it in a file.

7. Evaluate User Mood:

a. Use conditional statements to categorize the user's mood into different levels.

b. Provide personalized messages based on the determined mood level.

- If mood >= 3.5, print a positive message and offer stress relief games.

- If mood >= 3.0, print a positive message and offer stress relief games.

- If mood >= 1.5, acknowledge challenges, offer support, and optionally provide helpline information.

- If mood >= 0.5, acknowledge low mood, offer support, and optionally provide helpline information.

8. Stress Relief Games:

If the user chooses to play stress relief games, provide links to two websites.

9. Helpline Information:

a. If the user's mood indicates potential distress, inquire about persistent feelings.

b. Offer information on helpline numbers if the user expresses interest.

10. Positive Closure:

a. Conclude the interaction with positive messages.

b. Output reminders for self-love and well-being.

c. Wish the user the best.

11. End of Program:

Return 0 to indicate successful program execution.

SOURCE CODE

```
#include <iostream>
#include <fstream>
#include <string>
#include <limits>

using namespace std;

class User
{
private:[]
    string userName;
    string gender;
    int age;
    double mood;

public:
    User() : age(0), mood(0.0) {}

    void getUserInfo()
    {
        cout << "What is your name?\n";
        getline(cin, userName);

        cout << "\nHello, " << userName << "!\n";
```

```
    while (true)
    {
        cout << "What is your gender? (Male/Female/Other)\n";
        getline(cin, gender);

        for (char& c : gender)
        {
            c = tolower(c);
        }

        if (gender == "male" || gender == "female" || gender ==
"other")
        {
            break;
        }

        else
        {
            cout << "Invalid gender input. Please enter a valid gender
(Male/Female/Other).\n";
        }
    }

    while (true)
    {
        cout << "\nHow old are you?\n";
```

```
        if (cin >> age && age > 10 && age < 100)
        {
            break;
        }

        else
        {
            cout << "Invalid input. Please enter a valid age.\n";
            cin.clear();
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
        }
    }

    while (true)
    {
        cout << "\nOn a scale of 0.5 to 5, how are you feeling, " << userName << "?\n(A mood scale is provided below for your reference.)\n";
        cout << "Mood Scale:\n";
        cout << "0.5: Extremely Unhappy\n";
        cout << "1.0: Very Sad\n";
        cout << "1.5: Unhappy\n";
        cout << "2.0: Somewhat Down\n";
        cout << "2.5: Neutral\n";
        cout << "3.0: Content\n";
        cout << "3.5: Happy\n";
        cout << "4.0: Very Happy\n";
```

```
            cout << "4.5: Delighted\n";
            cout << "5.0: Ecstatic\n";

            if (cin >> mood && mood >= 0.5 && mood <= 5.0)
            {
                break;
            }

            else
            {
                cout << "Invalid input. Please enter a valid mood score.\n";
                cin.clear();
                cin.ignore(numeric_limits<streamsize>::max(), '\n');
            }
        }
    }

    void writeToFile() const
    {
        ofstream responseFile("user_responses.txt", ios::app);
        if (responseFile.is_open())
        {
            responseFile << "User: " << userName << ", Gender: " <<
gender << ", Age: " << age << ", Mood: " << mood << "\n";
            responseFile.close();
        }
```

```
        else
        {
            cerr << "Unable to open the response file for writing.\n";
        }
    }

    double getMood() const
    {
        return mood;
    }

    const string& getUserName() const
    {
        return userName;
    }
};

int main() {
    User user;
    user.getUserInfo();
    user.writeToFile();

    double mood = user.getMood();
    const string& userName = user.getUserName();

    if (mood >= 3.5)
    {
```

```
        cout << "\nThat's wonderful! We're genuinely happy for you and wish you continued joy and success in every aspect of your life. Always remember that you are enough. You are loved.\n";


        char gresponse;
            cout << "Would you like to play some stress relief games?(y/n)\n";
            cin >> gresponse;


            if(gresponse == 'y' || gresponse == 'Y')
            {
                cout << "1. [Silver Games - Stress Relief Games] - A website which provides you with stress relief games(You can copy and paste the link):\n";
                cout << "https://www.silvergames.com/en/t/stress-relief\n";
                cout << "2. [A Soft Murmur - Create Your Own Ambient Soundscape] - A website which lets you create your own soundscape and gives you a calming effect(You can copy and paste the link):\n";
                cout << "https://asoftmurmur.com/\n";
            }


            else
            {
                cout << "Cool! We hope you always remember to enjoy the little joys of life and embrace your inner child.";
            }
    }


    else if (mood >= 3.0)
```

```
    {
        cout << "\nBeing content is a beautiful state of mind. We hope this feeling persists and that you find utmost happiness in your life's journey. Always remember that you are enough. You are loved.\n";


        char gresponse;
            cout << "Would you like to play some stress relief games?(y/n)\n";
            cin >> gresponse;


            if(gresponse == 'y' || gresponse == 'Y')
            {
                cout << "1. [Silver Games - Stress Relief Games] - A website which provides you with stress relief games(You can copy and paste the link):\n";
                cout << "https://www.silvergames.com/en/t/stress-relief\n";
                cout << "2. [A Soft Murmur - Create Your Own Ambient Soundscape] - A website which lets you create your own soundscape and gives you a calming effect(You can copy and paste the link):\n";
                cout << "https://asoftmurmur.com/\n";
            }


            else
            {
                cout << "Cool! We hope you always remember to enjoy the little joys of life and embrace your inner child.";
            }
    }
```

```
    else if (mood >= 1.5)
    {
        cout << "\nWe understand that things might be tough. ";


        char response;
        cout << "Have you been feeling this way consistently over a period of time? (y/n)\n";
        cin >> response;


        if (response == 'y' || response == 'Y')
        {
            cout << "We want you to know that your feelings are valid. It's completely okay to feel this way. If this feeling persists over a prolonged period of time, consider reaching out to mental health professionals for support.\n";
            char response;
            cout << "Would you like information on helpline numbers? (y/n)\n";
            cin >> response;


            if(response == 'y' || response == 'Y')
            {
                cout << "For your reference, here are a few helplines that may provide assistance.\n1. Vandrevala Foundation Helpline: 1-860-266-2345\n2. Roshni Helpline: 040-66202000\n3. Snehi Helpline: 91-22-27719595 / 91-22-27719596\n4. iCall Helpline: 9152987821 / 9152987823\n5. Sumaitri Helpline: 011-23389090\n6. VIMHANS (Vidyasagar Institute of Mental Health and Neurosciences) Helpline: 011-66206620\n7. Parivarthan Counseling Helpline (Bangalore) Helpline: 7676602602\n8. COOJ Mental Health Foundation (Goa)
```

```
Helpline: 0832-2252525\n9. Samaritans Mumbai Helpline: 91-8422984528 / 91-8422984529 / 91-8422984530\n";
        }

        else
        {
            cout << "It's completely okay if you're not ready for helpline numbers right now. Remember, seeking help is a personal journey, and when you're ready, support will be available. Don't lose hope; everything will be fine. Take your time and reach out when you feel comfortable.\n";
        }

    }

    else
    {
        cout << "If this feeling is related to a specific incident, remember that setbacks happen to everyone. Always remember that you are enough. You are loved. Stay positive and focus on the things that bring you joy.\n";
    }

    char gresponse;
        cout << "Would you like some stress relief games as a way to take your mind off things?(y/n)\n";
        cin >> gresponse;

        if(gresponse == 'y' || gresponse == 'Y')
```

```
        {
            cout << "1. [Silver Games - Stress Relief Games] - A website which provides you with stress relief games(You can copy and paste the link):\n";
            cout << "https://www.silvergames.com/en/t/stress-relief\n";
            cout << "2. [A Soft Murmur - Create Your Own Ambient Soundscape] - A website which lets you create your own soundscape and gives you a calming effect(You can copy and paste the link):\n";
            cout << "https://asoftmurmur.com/\n";
        }

        else
        {
            cout << "Cool! We hope you always remember to enjoy the little joys of life and embrace your inner child.";
        }
    }

    else if (mood >= 0.5)
    {
        cout << "It's okay to feel this way. We're here for you. ";

        char response;
        cout << "Have you been feeling this way consistently over a period of time? (y/n)\n";
        cin >> response;

        if (response == 'y' || response == 'Y')
```

```
    {
        cout << "If you've been feeling consistently low over a period of time, it might be more serious than you think. Consider reaching out to mental health professionals for support. You are not alone.\n";
        char response;
        cout << "Would you like information on helpline numbers? (y/n)\n";
        cin >> response;

        if(response == 'y' || response == 'Y')
        {
            cout << "For your reference, here are a few helplines that may provide assistance.\n1. Vandrevala Foundation Helpline: 1-860-266-2345\n2. Roshni Helpline: 040-66202000\n3. Snehi Helpline: 91-22-27719595 / 91-22-27719596\n4. iCall Helpline: 9152987821 / 9152987823\n5. Sumaitri Helpline: 011-23389090\n6. VIMHANS (Vidyasagar Institute of Mental Health and Neurosciences) Helpline: 011-66206620\n7. Parivarthan Counseling Helpline (Bangalore) Helpline: 7676602602\n8. COOJ Mental Health Foundation (Goa) Helpline: 0832-2252525\n9. Samaritans Mumbai Helpline: 91-8422984528 / 91-8422984529 / 91-8422984530\n";
        }
        else
        {
            cout << "It's completely okay if you're not ready for helpline numbers right now. Remember, seeking help is a personal journey, and when you're ready, support will be available. Don't lose hope; everything will be fine. Take your time and reach out when you feel comfortable.\n";
        }
    }
```

```
    else
    {
        cout << "If this feeling is related to a specific incident, remember that it's okay to seek help. Consider talking to someone you trust or reaching out to helplines. Always remember that you are enough. You are loved.\n";

        char response;
        cout << "If you need someone to talk to and vent, we can provide the information of some helpline numbers. Would you like for us to provide you with it? (y/n)\n";
        cin >> response;

        if(response == 'y' || response == 'Y')
        {
            cout << "For your reference, here are a few helplines that may provide assistance.\n1. Roshni Helpline: 040-66202000\n2. iCall Helpline: 9152987821 / 9152987823\n3. Sumaitri Helpline: 011-23389090\n4. COOJ Mental Health Foundation (Goa) Helpline: 0832-2252525\n5. Samaritans Mumbai Helpline: 91-8422984528 / 91-8422984529 / 91-8422984530\n";
        }
        else
        {
            cout << "It's completely okay if you're not ready for helpline numbers right now. Remember, seeking help is a personal journey, and when you're ready, support will be available. Don't lose hope; everything will be fine. Take your time and reach out when you feel comfortable.\n";
        }
    }
```

```
        char gresponse;
            cout << "Would you like some stress relief games as a way to take your mind off things?(y/n)\n";
            cin >> gresponse;
            if(gresponse == 'y' || gresponse == 'Y')
            {
                cout << "1. [Silver Games - Stress Relief Games] - A website which provides you with stress relief games(You can copy and paste the link):\n";
                cout << "https://www.silvergames.com/en/t/stress-relief\n";
                cout << "2. [A Soft Murmur - Create Your Own Ambient Soundscape] - A website which lets you create your own soundscape and gives you a calming effect(You can copy and paste the link):\n";
                cout << "https://asoftmurmur.com/\n";
            }
            else
            {
                cout << "Cool! We hope you always remember to enjoy the little joys of life and embrace your inner child.";
            }
    }
    cout << "\nPlease always remember to love yourself first. Take good care of yourself. We wish you the best!";

    return 0;
}
```

CONCLUSION

This C++ program was designed with a human-centric approach, aiming to address the well-being of users in a supportive and interactive manner. Several key reasons underline the significance and purpose of this code:

1. User Engagement and Empathy:

- The program fosters a connection with users by engaging them in a friendly and conversational manner.

- Empathetic responses cater to users' emotional states, promoting a sense of understanding and support.

2. Information Collection for Record-Keeping:

- Gathering user information allows for a record of interactions, aiding in understanding user trends and needs over time.

- The "user_responses.txt" file serves as a valuable log for future reference and analysis.

3. Tailored Support Based on Mood:

- The program delivers personalized messages according to the user's reported mood, offering encouragement, understanding, and relevant resources.

- Mood-based differentiation ensures a nuanced and customized user experience.

4. Encouragement of Self-Care:

- Positive affirmations and reminders to prioritize self-love contribute to a positive and uplifting user experience.

- Stress relief game suggestions offer practical avenues for relaxation and enjoyment.

5. Promotion of Mental Health Awareness:

- In cases of lower mood scores, the program introduces the concept of seeking professional help, promoting mental health awareness and destigmatizing seeking assistance.

7. Versatility and Adaptability:

- The program's modular and well-organized code allows for easy maintenance, updates, and potential expansion of features.

- It provides a foundation for future enhancements and adaptations to meet evolving user needs.

8. Creators' Personal Values:

As the creators, we think the program reflects a commitment to user well-being, compassion, and the belief that technology can be harnessed for positive and meaningful interactions.

In essence, this C++ program is a manifestation of our vision to create a supportive and user-friendly environment, leveraging technology to connect with individuals on a personal level. It represents a blend of technical proficiency and a genuine desire to contribute positively to users' lives.