Umna Ellahi

SoulSync

Quick Submit

Quick Submit

National University of Computer and Emerging Sciences, Islamabad

Document Details

Submission ID
trn:oid:::1:3420692971

Submission Date
Nov 22, 2025, 10:47 AM GMT+5

Download Date
Nov 22, 2025, 10:51 AM GMT+5

File Name
SoulSync_Report.docx

File Size
22.3 KB

8 Pages

1,358 Words

8,103 Characters

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

Final Project Report — Programming Fundamentals

University Name: National University of Computing and Emerging Sciences

Department: Department of Cyber Security

Course: Programming Fundamentals

Project Title: SoulSync

Submitted By: Umna Ellahi 25K-2016

Submitted To: Sandesh Kumar

Semester: Fall 2025

Date: 21 November 2025

Abstract

Soul Sync is a mood journaling software that runs on a console utilizing the C programming language. This project aims to allow users to reflect on their moods daily, write short stories, and summarize trends of their emotional patterns over a specified period of time. The system also displays motivational quotes based on the user's selected mood. It also stores entries using arrays and file handling, performs calculations to show mood distribution through percentages. Through this project, I explored and applied programming concepts such as loops, functions, arrays, decision structures and simple file operations. To conclude, Soul Sync acts as a practical and easy to use journaling tool that combines basic dairy functions with simple mood-tracking intelligence.

1. Introduction

Paper diaries are often hard to organize and search through, and analyze; therefore, users cannot easily track patterns. Most of the existing digital journaling solutions are complex and require internet connectivity, and are cluttered with unnecessary features that overwhelm the user seeking only to simple experience.

SoulSync provides an easy-to-use interface with which users can add daily entries, record moods, and helping the users understand their emotional trends. The data is stored via arrays and file handling, safeguarding user confidentiality while permitting the user to see earlier entries and moods. Moreover, the system will show quotes appropriate to the mood chosen in order to keep the user engaged and motivated. This project acts as a more effective way of connecting more sophisticated digital models with more conventional paper diaries allowing the user an experience of thoughtful reflection about their day.

2. Objectives

- To develop an intuitive and highly usable interface for the user.
- To enable easy data management, such as the addition of entries and viewing by the user.
- To implement a feature for discrete mood logging corresponding to each user entry.
- To design an analytical module to compute the statistical data of recorded moods over a certain period.
- To deploy a dynamic quote generation system that displays motivational quotes that cater to any chosen mood by the user.
- A simple security protocol that verifies a password is implemented to protect the confidentiality of data.

3. System Design

System Overview

Algorithm

1. Start
2. Prompt the user to authenticate themselves through password verification.
3. Display the main menu interface.
4. Ask the user to input their choice.
5. Action processing loop:

- Mood Logging: record the user's emotional state and produce a corresponding motivational quote
- Journal Entry: record user's daily journal entry data
- Entry Retrieval: display recorded entries and facilitate user selection for review
- Mood Analysis: prompt the user to enter a period for analysis, then calculate and display the quantitative mood distribution in percentages
- Settings Management: display system settings and allow the user to modify or reset their current password.

6. After an action is completed, redisplay the main menu, prompting the user for their next action.

7. If the "Exit Program" option is selected then term the program should terminate, otherwise, go to Step 4

8. End

Input & Output

Input: User selection from the Main Menu

Output: (Journal Entry) Confirmation message upon the successful save of the user's journal entry

4. Implementation

Language: C

Compiler: Dev C++

Key Features

- **Data Security:** Password verification system to ensure user data confidentiality.
- **Mood Documenting:** Integration of a function to record the user's mood.
- **Journal Entry Creation:** The ability to add new entries to a journal
- **Entry Retrieval:** Enables the browsing and retrieval of earlier entries.
- **Quantitative Analysis:** Statistical distribution-based mood analytics.
- **Interface design:** Development of a menu-driven console interface
- **Contextual Feedback:** It generates inspirational quotes that are relevant to the mood that is captured.

Code Snippet

```
void mood_analytics() {
   int happyCount, sadCount, angryCount, neutralCount, excitedCount;
   int totalDays;
   printf("=== SMART DIGITAL DIARY - MOOD ANALYTICS ===\n\n");
   // Input section
   printf("Enter number of days in the selected range: ");
   scanf("%d", &totalDays);
   if (totalDays <= 0) {
      printf("\nError: Number of days must be greater than zero.\n");
      return;
   }
   printf("\nEnter mood counts:\n");
   printf("Happy entries   : ");
   scanf("%d", &happyCount);
   printf("Sad entries     : ");
   scanf("%d", &sadCount);
   printf("Angry entries   : ");
```

```
    scanf("%d", &angryCount);
    printf("Neutral entries : ");
    scanf("%d", &neutralCount);
    printf("Excited entries : ");
    scanf("%d", &excitedCount);
    int totalEntries = happyCount + sadCount + angryCount + neutralCount + excitedCount;
    // Validation: total entries must not exceed days
    if (totalEntries > totalDays) {
        printf("\nError: Total entries (%d) cannot exceed number of days (%d).\n", totalEntries,
totalDays);
        return;
    }
    // Calculation
    double happyProb   = (double)happyCount / totalDays * 100;
    double sadProb     = (double)sadCount / totalDays * 100;
    double angryProb   = (double)angryCount / totalDays * 100;
    double neutralProb = (double)neutralCount / totalDays * 100;
    double excitedProb = (double)excitedCount / totalDays * 100;
    // Output section
    printf("\n========================================\n");
    printf("         MOOD ANALYTICS REPORT         \n");
    printf("========================================\n");
    printf("Days in Range     : %d\n", totalDays);
    printf("Total Entries     : %d\n", totalEntries);
    printf("----------------------------------------\n");
    printf("Happy    : %d entries | Probability = %.2f%%\n", happyCount, happyProb);
    printf("Sad      : %d entries | Probability = %.2f%%\n", sadCount, sadProb);
    printf("Angry    : %d entries | Probability = %.2f%%\n", angryCount, angryProb);
    printf("Neutral  : %d entries | Probability = %.2f%%\n", neutralCount, neutralProb);
```

```
    printf("Excited  : %d entries | Probability = %.2f%%\n", excitedCount, excitedProb);
    printf("=========================================\n");
}
```

Sample Output

```
=== SMART DIGITAL DIARY - MOOD ANALYTICS ===

Enter number of days in the selected range: 20

Enter mood counts:
Happy entries   : 9
Sad entries     : 1
Angry entries   : 2
Neutral entries : 8
Excited entries : 0

=========================================
        MOOD ANALYTICS REPORT
=========================================
Days in Range     : 20
Total Entries     : 20
-----------------------------------------
Happy    : 9 entries | Probability = 45.00%
Sad      : 1 entries | Probability = 5.00%
Angry    : 2 entries | Probability = 10.00%
Neutral  : 8 entries | Probability = 40.00%
Excited  : 0 entries | Probability = 0.00%
=========================================
```

5. Testing & Results

Test No	Input	Expected Output	Actual Output	Status
1	Days: 30 Mood:10,10,10,5,1	Error Message	Total entries (36) cannot exceed number of days(30)	✓
2	Days: 30 Mood: 8,6,4,7,5	Valid Probabilities	Correct Calculation E.g H = 26.67%	✓
3	Days: 30 Mood: 6,6,6,6,6	Prob = 20% for all moods	20%	✓

The program performed successfully for all test cases. It handled both wrong, neutral, and general case moods efficiently and produced accurate outputs while validating all inputs. Its execution speed was nearly instantaneous, and the program required minimal system resources.

6. Conclusion, Limitations & References

Conclusion

SoulSync does indeed provide a very simple and clean console-based journaling system. It shows how fundamental programming concepts such as loops, arrays, functions, and file handling can be applied to demonstrate a clean, practical, and interactive software. It showcases the traditional paper diary but with extended functionality such as mood tracking using probability; relating moods to entries, pattern analysis over a selected time range, and motivational quotes based on the selected mood. It acts not only as a simple diary but also as a means for self-reflection tool and mental health companion in this fast-moving world.

Limitations

- The program cannot currently import or export any files.
- Image attachment is not supported by the system.
- This application also only operates as a command-line application and does not have a graphical user interface (GUI) and as such is not visually appealing to the user.
- Data is not encrypted at the time of writing, thereby putting the user's privacy at risk.
- The application does not allow the user to delete an existing entry and limits user options in that respect.

Future Enhancements

- Shift to a Graphical User Interface (GUI).
- Provide a strong data encryption for security and the privacy of the user's entries.
- Introduce graphical representations of mood analytics and long-term trends.
- Design a cloud-based backup solution in order to achieve persistence of data across several devices.
- Allow the user to import and export external files and images within each journal entry.
- The program will provide an option to delete an entry; this gives more choice to the users.
- Implementing User Activity Tracking System which will allow the user to keep a track of their activity

References

- Google Images
- https://www.programiz.com/c-programming