Project Report: Diary Management System

Md Abul Bashar Nirob - 2022198042
Jannatul Islam Eshita - 2524186042
Tashin Binte Taiba - 2522283642
Sara Tasnim - 2522727042
Tasnia Afrin - 2523855642

Course Code: cse-115, Project Group Number: 7
July 05, 2025

Abstract—Creating a personal diary management system involves developing a tool that allows users to record, manage, and secure their daily thoughts and experiences. This report outlines the progress made on a C-based application designed to handle diary entries with features such as adding, viewing, searching, editing, deleting, encrypting, decrypting, and searching for related items. The project also includes an interface with color-coded menu options for enhanced usability.

I. Introduction

Creating a personal diary management system involves developing a tool that allows users to record, manage, and secure their daily thoughts and experiences. This report outlines the progress made on a C-based application designed to handle diary entries with features such as adding, viewing, searching, editing, deleting, encrypting, decrypting, and backing up entries. The system aims to provide a user-friendly interface with color-coded menu options for enhanced usability.

II. System Design and Implementation

The diary management system is implemented using C programming language, leveraging file handling for data persistence. Key components include:

- Data Storage: Entries are stored in a text file ("diary.txt") with a simple format of date and entry pairs.
- Functionality: The system includes eight main functions-addEntry, viewEntries, searchEntry, editEntry, deleteEntry, encryptFile, decryptFile, and backupDiary-each handling a specific task.
- Security: Encryption and decryption features use a basic XOR cipher with a user-defined password, stored in a separate encrypted file.
- User Interface: A color-coded menu system enhances user interaction, utilizing Windows console functions for visual feedback.

III. Progress and Challenges

Significant progress has been made in developing the core functionalities. The addEntry and viewEntries functions are fully operational, allowing users to input and review entries. The searchEntry function successfully locates entries by date, while editEntry and deleteEntry use temporary files to modify or remove data effectively. Encryption and decryption are implemented, though password verification needs further refinement to prevent unauthorized access. The backupDiary function creates timestamped backups, ensuring data safety. Challenges include ensuring robust error handling for file operations and improving the encryption algorithm's security. Initial tests revealed occasional segmentation faults during file opening, which were resolved by adding proper file closure checks.

IV. Results and Testing

Testing was conducted to evaluate the system's performance. The addEntry and viewEntries functions successfully processed over 50 entries, with data accurately saved and retrieved from "diary.txt". The searchEntry function located entries within 2 seconds for a file with 30 entries, while editEntry and deleteEntry maintained data integrity during modifications. Encryption and decryption tests showed successful file protection with correct passwords, though incorrect passwords failed to decrypt, indicating a need for better error messaging. The backupDiary function created a portable backups, verified by comparing file contents. Further testing is planned to address edge cases like file corruption.

V. Future Work

Future updates will focus on enhancing security by implementing a more robust encryption method, such as AES. Additional features like entry categorization and a graphical user interface (GUI) using a library like ncurses are planned. Testing will be expanded to include edge cases, such as large file sizes and invalid inputs, to ensure reliability.

VI. Conclusion

The diary management system demonstrates a functional prototype, with essential features for personal record-keeping. Continued development will address current limitations and add advanced functionalities, aligning with the project's goal of providing a secure and efficient tool.

References

[1] C Programming tutorials on file handling, available at https://www.wdschools.com/c/c/yiles.php.

[2] General guidance on software design from academic resources and peer discussions.