Personal Diary Management System: A Structured Approach

Fabiha Raiyan
2421170642

Aminur Rashid Sohag
2514160042

Tanvir Ahmed Ovi
2512723642

Imtiaj Ahmed
2513119042

Abstract

The Personal Diary Management System is designed to provide a structured and secure method of storing personal diary entries. This program implements fundamental C programming concepts such as file handling, input validation, user authentication, encryption using XOR, and structured data storage. The current implementation supports entry addition, viewing, editing, and deletion while ensuring that duplicate dates are not allowed. This report outlines the functionality, structure, and future enhancements of the system.

1. Introduction

A personal diary is a significant way of recording thoughts, events, and experiences. The Personal Diary Management System enhances this process by digitizing diary entries, ensuring security through authentication, and managing data efficiently using structured file handling. The system is developed in C and follows a modular approach for better readability and maintainability.

2. System Features

The system includes the following key features:

- **User Authentication:** A PIN-based authentication system ensures secure access.
- **Date Validation:** Entries must follow the format YYYY-MM-DD and pass a validation check.
- **Diary Entry Management:** Users can add, view, edit, and delete diary entries.
- **Duplicate Entry Prevention:** The system does not allow multiple entries for the same date.
- **Encryption Using XOR:** Entries are encrypted before being stored for added security.
- **File Handling:** Entries are stored in a structured file format to ensure data persistence.

3. Implementation

The system is implemented using standard C programming techniques. Key components include:

3.1. User Authentication

A simple function authenticate() verifies user access via a predefined PIN. Users must enter the correct PIN before proceeding to the main menu.

3.2. Date Validation

The function isValidDate(y, m, d) checks whether the entered date is valid. It ensures:

- The month is between 1 and 12.
- The day is within the valid range for the respective month.
- February can have up to 29 days if it is a leap year.

3.3. Diary Entry Management

- addEntry(): Accepts a valid date and diary content while preventing duplicate entries.
- viewEntry(): Displays all stored entries in chronological order.
- editEntry(): Enables users to modify an existing entry.
- deleteEntry(): Removes an entry from storage if it exists.
- encrypt(): Uses XOR-based encryption to protect diary content before saving.

4. Future Enhancements

The current implementation lays the foundation for a robust diary management system. Future enhancements include:

- **Advanced Encryption:** Implementing a stronger encryption algorithm for enhanced security.
- **Optimized File Handling:** Using indexing or database-like structures for better efficiency.

- **Improved User Interface:** Enhancing the user experience with better navigation and interaction features.

5. Conclusion

The Personal Diary Management System is a structured approach to digital diary keeping. It ensures security through authentication, prevents data duplication, and enforces valid date input. The addition of XOR encryption enhances security, while file handling ensures data persistence. While the system is currently functional, further improvements will enhance usability and security. This project serves as a learning tool for applying fundamental C programming concepts in a real-world application.

6. References

[1] Briand, L. C., Daly, J., and Wüst, J., "A unified framework for coupling measurement in object-oriented systems," *IEEE Transactions on Software Engineering,* 25(1), January 1999, pp. 91-121.

[2] Maletic, J. I., Collard, M. L., and Marcus, A., "Source Code Files as Structured Documents," in *Proceedings 10th IEEE International Workshop on Program Comprehension*, Paris, France, June 2002, pp. 289-292.