Personal Library Management System (PLMS)

CSCE3102 - Programming in Java (Fall 2024)

Ahmed Badr (900202868) - ahmedbadr00@aucegypt.edu
Bemen Girgis (900213066) - bemen@aucegypt.edu

December 6, 2024

1 Project Overview

The Personal Library Management System (PLMS) is a comprehensive Java application designed to help users manage their book collections efficiently. The system implements a graphical user interface for book management operations while utilizing advanced Java features such as Collections Framework, file handling, and object serialization.

2 Technical Implementation

2.1 Design Choices

2.1.1 Data Structures

- Implemented `HashMap` for ISBN-based book storage, ensuring O(1) access time
- Utilized `TreeMap` for genre-based categorization, maintaining sorted order
- Employed `TreeSet` for maintaining sorted collections
- Deliberately avoided ArrayList as per project requirements, opting for LinkedList where sequential access was needed

2.1.2 GUI Implementation

- Built using Java Swing framework
- Implemented Grid layout for form inputs to ensure consistent spacing
- Utilized JTable for book display with sorting capabilities
- Implemented dynamic updates for all operations
- Added responsive feedback through dialog boxes

2.1.3 File Handling

- Implemented buffered operations for improved I/O performance
- Developed custom serialization format with version control
- Added CSV support for data portability and backup
- Implemented automatic save on application close

3 Features

3.1 Core Features

1. **Book Management**
    - Add books with comprehensive details
    - Remove books by ISBN or table selection
    - Search functionality by Title, Author, or ISBN
    - Dynamic table display with sorting capabilities
2. **Categorization**
    - Genre-based grouping using TreeMap
    - Dynamic filtering system
    - Automatic genre list updates
3. **File Persistence**
    - Automatic save functionality
    - Load on startup
    - Manual load option

3.2 Advanced Features

1. **Serialization**
    - Custom serialization mechanism
    - Version control implementation
    - Robust error handling
2. **File Operations**
    - CSV export functionality
    - CSV import with validation
    - Automatic backup system
3. **Sorting and Filtering**

- Multiple sorting options (Title, Author, Year, Genre)
- Genre-based filtering
- Combined sort and filter capabilities

4 Validation Implementation

4.1 ISBN Validation

- Strict 13-digit format requirement
- Uniqueness verification
- Required field validation

4.2 Book Data Validation

- Duplicate Title + Author combination prevention
- Year format validation
- Required fields verification
- Case-sensitive genre matching

5 Running Instructions

5.1 Prerequisites

- Java Development Kit (JDK) 17 or higher
- Operating System: Windows, Linux, or macOS

5.2 Compilation Commands

```
javac Book.java LibraryManager.java LibraryGUI.java
```

5.3 Execution Commands

```
java LibraryGUI
```

5.4 Testing Commands

```
javac LibraryTest.java
java LibraryTest
```

6 Error Handling

The system implements comprehensive error handling for:

- Invalid ISBN formats
- Duplicate book entries
- Invalid year formats
- File operation failures
- Data corruption scenarios
- CSV format inconsistencies

7 Limitations and Future Enhancements

7.1 Current Limitations

- No batch operation support
- Single file backup system
- No undo/redo functionality
- Case-sensitive genre matching

7.2 Potential Improvements

- Multiple file format support
- Batch import/export capabilities
- Advanced search functionalities
- Statistical analysis features
- Multi-user support
- Cloud backup integration

8 Conclusion

The Personal Library Management System successfully implements all required features while maintaining good software engineering practices. The system provides a robust, user-friendly interface for book management with comprehensive error handling and data persistence capabilities. While there are areas for potential improvement, the current implementation satisfies all core requirements and provides a solid foundation for future enhancements.