Assignment 1:

Submitted to:Dr. Javeria Kanwal

Submitted by: Abdul Waris(SP23704)

Course:SCD(BSSE-5)

Software Requirements Specification (SRS) Document

For Library Management System

1. Introduction

1.1 Purpose

The Library Management System (LMS) will be a desktop application to automate library operations, including book catalog management, member registration, loan tracking, and fine calculation.

1.2 Document Conventions

- **User Roles:** Staff (admin) and Members (users).
- Focuses on functional requirements without technical implementation details.

1.3 Intended Audience

- **Stakeholders:** Library staff, management.
- **Developers:** For system design.
- **Testers:** For validation.

1.4 Scope

In Scope:

- Book and member management.
- Loan tracking.
- Fine assignment.

Out of Scope:

- User authentication (initially).
- Persistent storage (v1.0).

2. Overall Description

2.1 Product Perspective

Standalone application with a GUI (no command-line interface).

2.2 User Classes

Staff Permissions:

- Add/remove books and members.

- Assign fines.
- View all loans.

Member Permissions:

- Browse and borrow books.
- Return books.
- Pay fines.

2.3 Operating Environment

- **Platform:** Java-based (Windows/macOS/Linux).
- **Hardware:** Minimal requirements (any modern PC).

3. System Features

3.1 Functional Requirements

Staff Features:

FR1: Add Book

- Input: Book title.
- Output: Confirmation message.

FR2: Remove Book

- Input: Book title.
- Output: Success/error message.

FR3: View All Books

- Input: None.
- Output: List of books with loan status.

FR4: Add Member

- Input: Member name.
- Output: Confirmation.

FR5: Remove Member

- Input: Member name.
- Output: Success/error message.

FR6: Assign Fine

- Input: Member name, fine amount.
- Output: Updated fine record.

Member Features:

FR7: Browse Books

- Input: None.
- Output: List of available books.

FR8: Borrow Book

- Input: Member name, book title.
- Output: Loan confirmation.

FR9: Return Book

- Input: Book title.
- Output: Return confirmation.

FR10: Pay Fine

- Input: Member name, payment amount.
- Output: Updated fine balance.

3.2 Non-Functional Requirements

NFR1: Usability

- Metric: Intuitive GUI (≤3 clicks per task).

NFR2: Performance

- Metric: Response time <1s for all operations.

NFR3: Reliability

- Metric: No data loss during session.

4. Interface Requirements

4.1 User Interfaces

- **Main Menu:** Role selection (Staff/Member).
- **Staff Dashboard:** Buttons for book/member management.
- **Member Dashboard:** Options to browse/borrow books.

4.2 Software Interfaces

- **Language:** Java (JDK 8+).
- **Libraries:** Java Swing (for GUI).

5. System Models (Planned)

5.1 Data Flow

- Staff adds book → System updates book list.
- Member borrows book → System records loan.

6. Risks & Mitigation

Risk 1: No user authentication.

- Mitigation: Implement in v2.0.

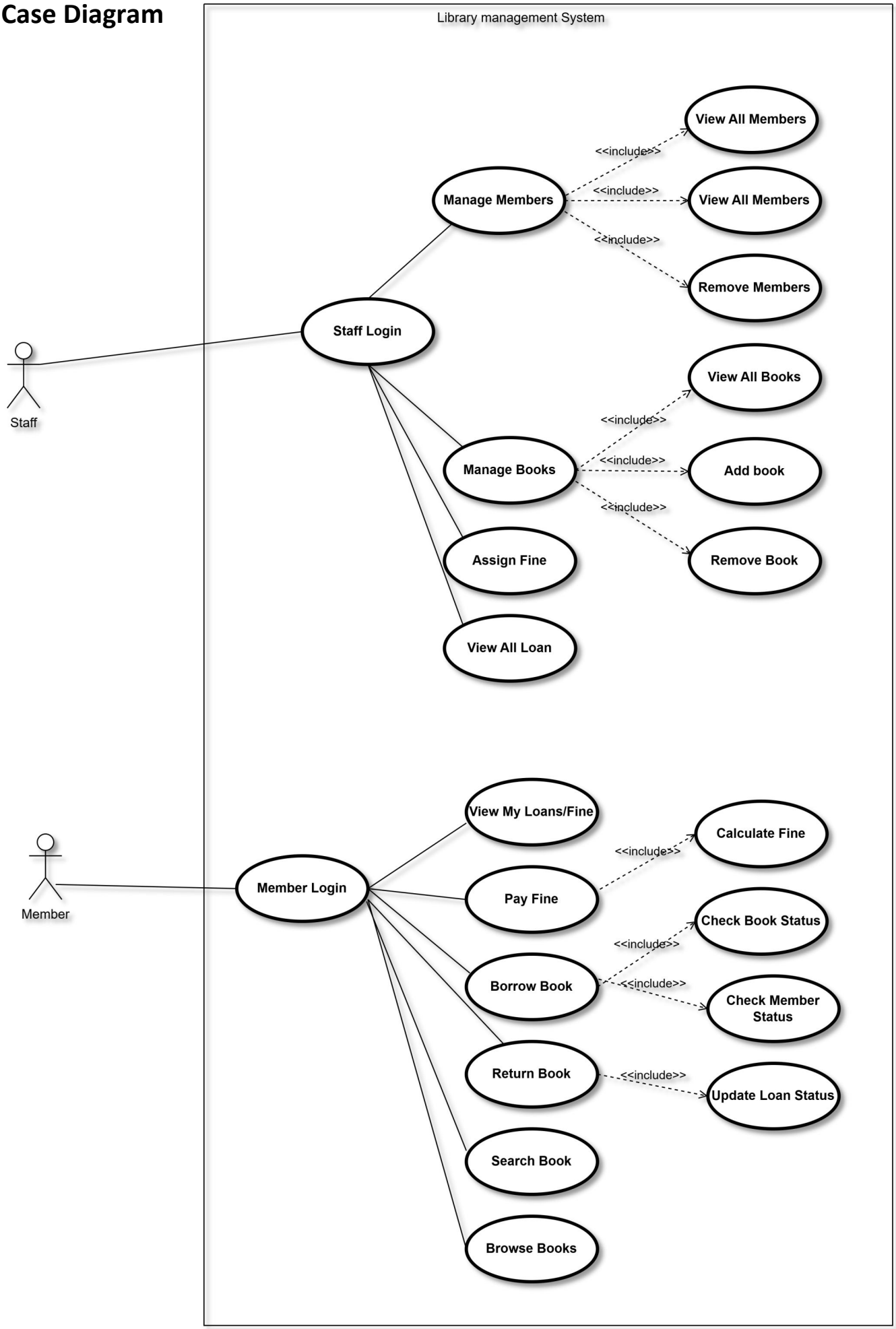
Risk 2: Data volatility (no database).

- Mitigation: Use file storage in future versions.

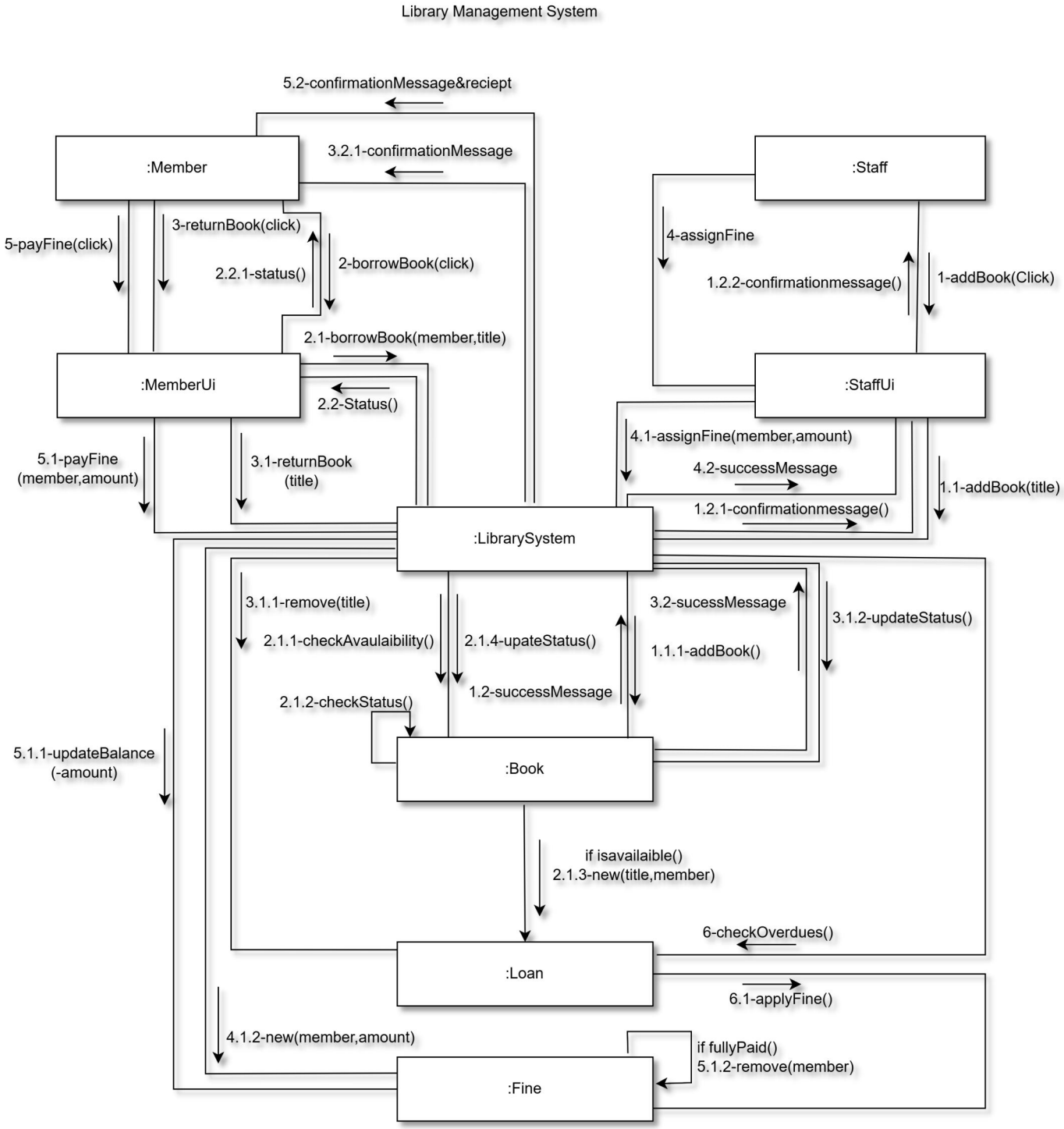
7. Future Enhancements

- Due date tracking for loans.
- Email notifications for overdue books.
- Multi-user login system.

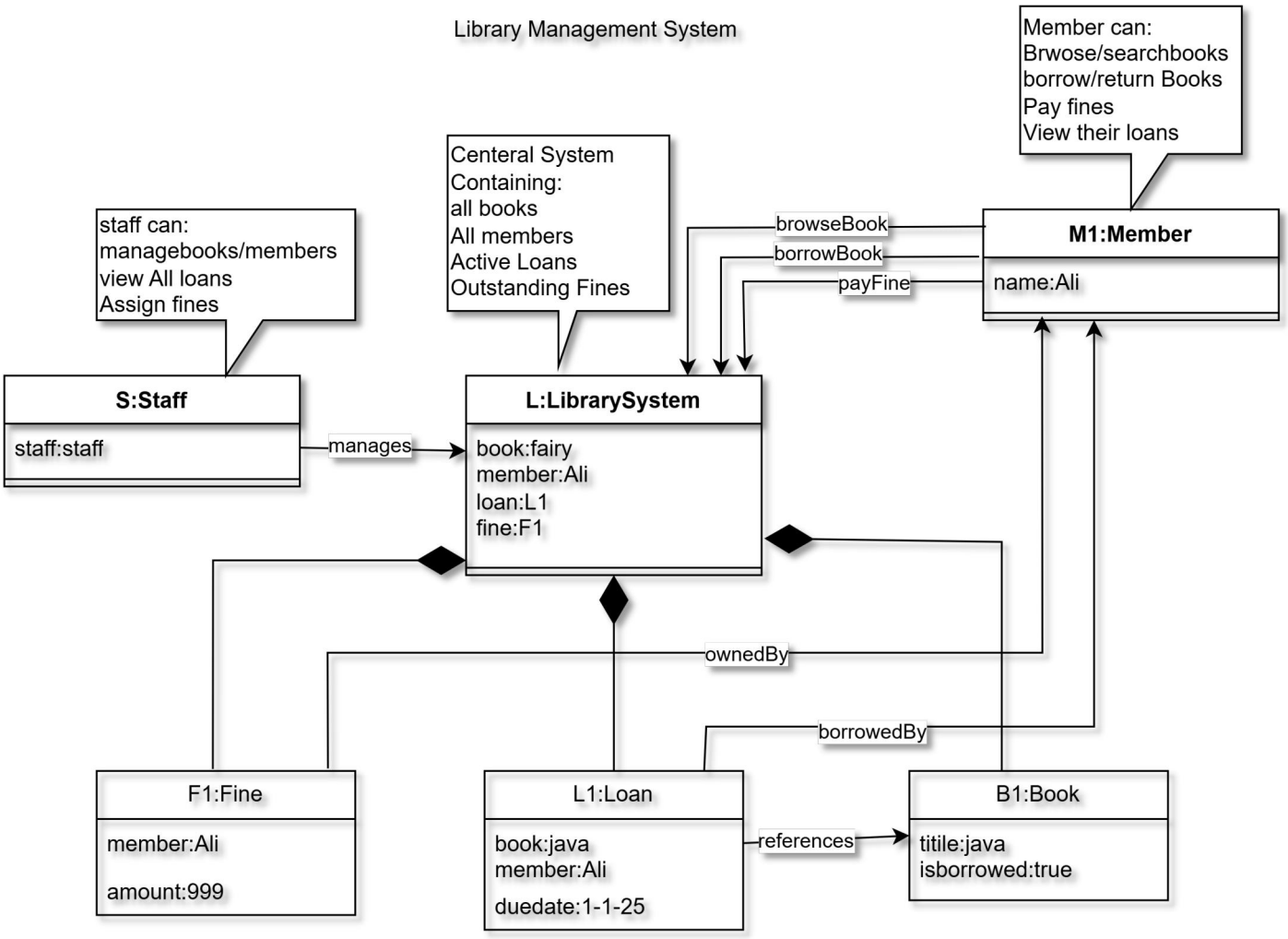
Detaied Design Document:
Use Case Diagram



Communication Diagram:



Object Diagram:



Sequence Diagram: