Software Design Document (SDD)
Library Management System

Meghana N Naik – 2023BCSE07AED584
CSE General

1. Introduction

The Software Design Document (SDD) describes the overall design and architecture of the Library Management System (LMS). The purpose of this document is to provide a detailed description of how the system is structured, how different components interact, and how data is processed and stored. This document serves as a blueprint for developers, testers, and stakeholders to understand the internal design of the system before implementation.

2. System Architecture

The Library Management System follows a **modular and layered architecture**, which separates the system into different functional components. This approach improves maintainability and makes the system easier to enhance in the future.

2.1 High-Level Architecture

The system is divided into the following major layers:

1. **Presentation Layer**
    - Provides the user interface for librarians and users
    - Includes login pages, dashboards, book search, and report views
    - Implemented using web technologies such as HTML, CSS, and JavaScript
2. **Application Layer (Business Logic Layer)**
    - Handles core functionality of the system
    - Processes requests such as book issue, return, fine calculation, and validations
    - Ensures business rules are applied correctly
3. **Data Access Layer**
    - Manages communication between the application and the database
    - Performs data retrieval, insertion, update, and deletion operations
4. **Database Layer**
    - Stores all persistent data such as books, users, transactions, and fines

2.2 System Modules

The major modules of the Library Management System include:

- **User Management Module – Manages librarian and user accounts**
- **Book Management Module – Handles book cataloging and availability**
- **Issue and Return Module – Controls book lending operations**
- **Fine Management Module – Calculates overdue fines**
- **Report Module – Generates usage and inventory reports**

Each module interacts with the database through the data access layer.

3. Detailed Design

Detailed design explains the internal logic, data flow, and interactions between system components.

3.1 Class Design

Key classes in the system include:

- **User** – Stores user details such as user ID, name, role, and contact information
- **Book** – Stores book details such as book ID, title, author, category, and status
- **Transaction** – Maintains issue and return details
- **Fine** – Stores fine amount and payment status

Each class contains attributes and methods required to perform its operations.

3.2 Data Flow Design

The data flow in the Library Management System follows a structured approach:

1. User submits a request through the user interface
2. Request is validated by the application layer
3. Required data is fetched or updated in the database
4. Response is returned to the user interface

For example, when a book is issued, the system checks availability, updates the transaction record, and modifies the book status.

3.3 Logic Description

- Login logic verifies user credentials before granting access
- Book issue logic checks availability and user eligibility
- Return logic calculates fine if the book is overdue
- Report logic retrieves summarized data from the database

These logical processes ensure correct and consistent system behavior.

4. Database Design

The database design defines how data is stored and managed in the Library Management System.

4.1 Entity Relationship Description

The main entities in the system are:

- **User** – Represents library members and librarians
- **Book** – Represents books available in the library
- **Transaction** – Represents book issue and return records
- **Fine** – Represents overdue penalties

Relationships:

- One user can issue multiple books
- One book can have multiple transaction records
- Each transaction may generate a fine

4.2 Data Storage Approach

- Relational database model is used
- Primary keys uniquely identify records
- Foreign keys maintain relationships between tables
- Indexing is applied to frequently accessed fields such as book ID and user ID

Regular backups are planned to prevent data loss.

5. Design Constraints

- Assign tasks based on team members' skills and strengths.
- Provide basic training where necessary.
- Maintain proper documentation to reduce dependency on individuals.
- Ensure effective communication through regular meetings.

5.1 Hardware Constraints

- System must run on standard computers
- Minimum hardware requirements include basic processor, RAM, and storage
- Server resources may be limited in small libraries

5.3 Software Constraints

- System must support common operating systems
- Database and development tools should be compatible with the platform
- Preference for open-source technologies to reduce cost

5.4 Regulatory and Operational Constraints

- User data privacy must be maintained
- System must follow institutional IT policies
- Access control must restrict unauthorized usage