

Design Documentation

A Proposed Library Management System at Navotas City Library

Prepared by: Caldit, Peter joel M.

BSIT 3d

INTRODUCTION

The purpose of this document is to describe the system design of the Library Management System. This project is part of our academic requirement, but more importantly, it addresses a real need we observed in our school community.

The Navotas City Library currently uses manual methods to record, track, and manage books, which often leads to errors, lost of records, and delays especially during busy periods. Through this system, we aim to improve those processes and provide a more organized, efficient, and user-friendly way to manage books and and track borrowed books and returned books.

Our system is designed as a desktop application using Visual Basic 2010 and Sql Server Management Studio making it accessible even without internet connection. It allows the librarians to manage books, monitor availability and location, and manage borrow and return books, all within a single platform. We focused on simplicity and functionality, ensuring that users with basic computer skills can operate it smoothly.

This document includes the system's architecture, database design, user interface layout, component breakdown, and security features. It also outlines how we plan to deploy and maintain the system after development. As students, we applied what we have learned in class from coding to proper documentation while keeping the system relevant and practical for its intended environment.

SYSTEM ARCHITECTURE

Architecture Type:

- Desktop-based, single-user system.

High-level Components:

- Client-side (Desktop Interface for Librarians)
- Built using Visual Basic 2010.
- Allows the librarian to manage books, and track borrowed books and returned books.

Server-side (System Logic & Operations)

- Handles book registration, updates, and availability tracking.
- Processes borrowing and returning operations.

Database (Sql Server Management Studio)

- Stores book information, availability status, and borrowed and returned records.
- Supports queries for reporting and data retrieval.

Deployment:

Communication Protocols and Interfaces:

- The system uses direct database queries to perform CRUD operations (Add, Update, Delete).
- All interactions between the user and the database are handled internally through Visual Basic forms and modules.
- Input validation and error handling are built into the interface to ensure data accuracy and prevent system crashes.

DATABASE DESIGN

The ERD shows main parts of the system:

Librarian – The one who is in control of the system.

Visitor – The one that visit's the library.

Relationship:

Visitors can borrow books.

Each request is process by librarians.

Description of Database Tables, Fields, and Relationships

Book:(ID, BookName, BookAuthorName, BookGenre, BookPurchaseDate, BookLocation, BookQuantity)

Borrow:(ID, BorrowerName, ContactNo, Place, BookName, BookBorrowDate, BookDueDate)

Data Normalization Techniques Used

To make the database efficient and avoid duplication of data, normalization techniques were applied up to Third Normal Form (3NF).

First Normal Form (1NF):

All tables have unique rows and no repeating groups.

Each field contains only simple and indivisible values.

Second Normal Form (2NF):

All non-key attributes depend fully on the primary key.

This removes partial dependency (when a field depends on only part of a composite key).

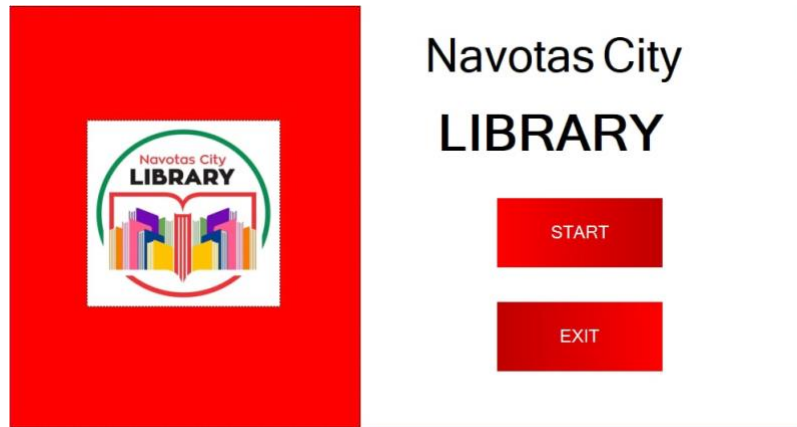
Third Normal Form (3NF):

All attributes are dependent only on the primary key, not on other non-key attributes. This removes transitive dependency.

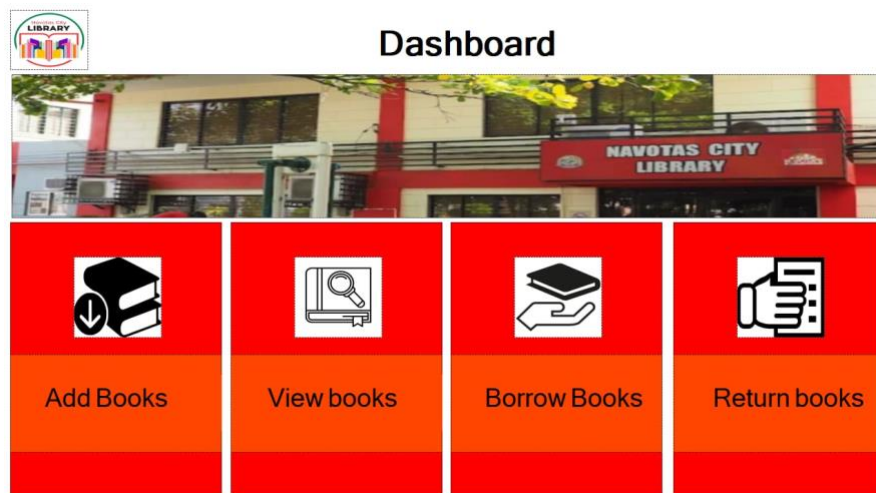
USER INTERFACE DESIGN

Wireframe

Starting Page



Dashboard



Add Book

Add Books



Book Name

Book Author Name

Book Genre

BookArrival Date

Book Location

Book Quantity

Tuesday, 2 September 2025

Add

Clear

Back

View Books

View books

Book Name

Refresh

Book Name

Book Authors Name

Book Genre

BookArrival Date

Book Location


Book Quantity

Update
Delete
Clear
Back

id

Borrow Book

Borrow Books



Borrower Name

Contact no.

Place

Book Name

Book Borrow Date

Book Due Date

Wednesday, 3 September 2025

Wednesday, 3 September 2025

Borrow

Clear

Back

Return Book

Return books

Borrower Name

Refresh

id

Borrower Name

Contact no.

Place

Book Name

Book Borrow Date

Book Due Date

Returned

Back

COMPONENT DESIGN

Starting Page

The starting page serves as the login screen It ensures that only authorized users can access the system.

Dashboard

The dashboard is the main control panel of the system. It provides quick access to key features such as adding books, viewing books, borrowing books and returning books.

Add Books Page

The add books page allows the librarian to register new books into the system by entering details such as title, author, and etc.

View Books Page

The view books page displays a list of all books stored in the system. The librarian can search, filter, or update book records from this page.

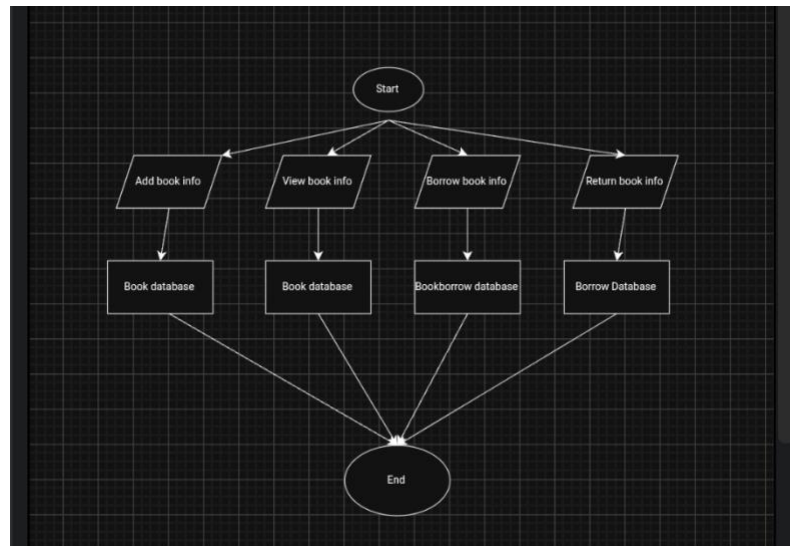
Borrow Book Page

The borrow book page enables the librarian to process book by inputting borrower's information.

Return Book Page

The return book page allows the librarian to record the return of borrowed books.

Data Flow Diagrams (DFD)



Data Sources

Add books – Details about books (Books name , Books author, quantity, and others.)

View books – Details about books that you can update and delete(Books name , Books author, quantity, and others.)

Borrow books- Details about borrowers information (Borrower name, Contact no. , Book name, and others.)

Return books - Details about borrowers information that you can delete(Borrower name, Contact no. , Book name, and others.)

Processing Logic

Start

System is activated or opened.

Add book

User inputs books details .

System stores this in the book database.

View book

User inputs books details .

System stores this in the book database.

Borrow book

User inputs books details .

System stores this in the bookborrow database.

Return book

User inputs books details .

System stores this in the book database.

End

Process finishes or system is closed.

Destination

Add book and view book is stored in the books database while borrow and return books is stored in bookborrow database.

Security Design

The Library Management System is designed to protect book and transaction data through built-in restrictions and controlled access. Since the system is desktop-based and operated only by librarians, security focuses on preventing data errors, unauthorized modifications, and accidental misuse of system functions. Input validation is applied to ensure that only correct and complete information is entered, while all actions are handled through verified system functions to avoid direct tampering with the database.

Authentication and Authorization Mechanisms

Authentication: The system does not include a login module, as it is intended for use exclusively by authorized librarians within the library premises.

Authorization: Only librarians are permitted to operate the system. Sensitive functions such as adding, updating, or deleting books, as well as processing borrow and return transactions, are restricted to librarian use.

Data Protection Measures

Input validation ensures that no incomplete or invalid data can be stored in the database

Book IDs and Bookborrow IDs are unique to prevent duplication or inconsistencies.

The database is secured within the system folder, with no direct external access, so all interactions occur only through the application interface.

Regular database backups are recommended to prevent data loss due to system errors or hardware failure.

Session and Access Control

Since the system is offline and single-user, no session management is required. Instead, physical security is ensured by limiting computer access to librarians only. This guarantees that library records remain protected and under librarian supervision at all times.

Performance Design

Performance Requirements and Objectives

The Library Management System must process core transactions such as adding books, updating records, borrowing, and returning within a few seconds to ensure smooth library operations. The database should be able to handle hundreds of book records and transaction logs without noticeable delays, as long as the computer has sufficient storage and memory.

Strategies for Optimizing System Performance

Optimized Queries: SQL queries in MS Access are structured to make searching and filtering of books faster.

Data Normalization: The database is normalized to remove redundancy and improve efficiency when storing books, users, and transaction records.

Efficient UI: Forms are designed with search and filtering options to avoid loading all records at once, ensuring quicker response times.

Resource Management: The system automatically closes unused forms and clears variables to free up memory during long sessions.

Performance Testing Plan

Load Testing: Simulate 50–100 borrow/return transactions in a day to evaluate system responsiveness.

Stress Testing: Populate the database with 1,000+ book records and measure search and update execution times.

Scalability Testing: Add continuous borrow/return requests to ensure the system remains stable as the number of transactions grows .

Error Handling and Logging

Error Handling Mechanisms

Input validation is applied on all forms to ensure that librarians cannot leave required fields blank (e.g., book title, author, borrower name) or enter invalid data types (e.g., letters in date fields).

User-friendly error messages are displayed to guide librarians in correcting mistakes instead of allowing the system to crash.

Automatic rollback is triggered if a database operation fails during a borrow or return process, ensuring data integrity.

Logging Requirements and Specifications

The system maintains an error log file recording all system errors, with details such as timestamp, action attempted, and error type.

A transaction log is kept for every successful borrow or return, including Transaction ID, Book ID, Borrower Name, and Date.

A book log tracks book-related actions such as adding, editing, or removing a book for accountability.

Logs are stored in the database for future reference and auditing by librarians.

Error Codes and Messages

E101 – Invalid Input: Displayed when a required field is left blank or the wrong data type is entered.

E202 – Book Not Available: Displayed when a librarian tries to borrow a book that has already been borrowed or marked as unavailable.

E303 – Book Record Not Found: Displayed when the system cannot find the book ID entered.

E404 – Database Connection Error: Displayed if the system fails to access the database.

E505 – Unauthorized Action: Displayed if an attempt is made to delete or modify records outside allowed operations.

Third-Party Integrations

List of Third-Party Services or Tools Integrated

MS Access Database Engine (ACE OLEDB): Used for database connectivity between the Visual Basic application and the MS Access database.

Optional Barcode Generation Library: Can be used to create barcodes or QR codes for books to make searching and borrowing faster (if implemented).

Integration Points and Data Exchange Formats

Database Connectivity: The system communicates with MS Access through SQL queries using the ACE OLEDB provider. All book records, borrower details, and transaction logs are stored and retrieved in a structured database format consisting of tables and queries.

Barcode Generator (if applied): Exports barcode images as PNG/JPG for book labeling and scanning, enabling easier book identification during borrowing and returning.

Data Exchange: All interactions remain within the structured database schema, ensuring consistency and integrity of library records.

Deployment Plan

Overview of the Deployment Process

The Library Management System will be deployed as a desktop-based application on the library's computer. The deployment process includes installing MS Access runtime and Visual Basic runtime libraries, copying the system executable (.exe file) to the local machine, and configuring the database path to ensure it can be accessed by the application. Once installed, all core modules (Add Books, View Books, Borrow, Return, and Reports) will be tested to verify proper functionality.

Hardware and Software Requirements for Deployment

Hardware:

Personal Computer with at least 4GB RAM, 250GB storage, and dual-core processor.

Printer (optional, for printing reports or book lists).

Software:

Windows 10/11 operating system.

MS Access 2016 or later (or Access Runtime version).

Visual Basic runtime libraries.

Configuration Management and Version Control Procedures

The system source code will be stored and maintained in a version control repository such as GitHub, enabling developers to track changes, fix issues, and implement updates in an organized manner. A version control naming convention (e.g., v1.0, v1.1, v1.2) will be followed to identify each release clearly.

To ensure data and system security, regular backups of both the system executable and the MS Access database will be performed. This ensures that, in the event of system errors, data corruption, or accidental deletion, the most recent working version of the system can be restored quickly.

Revision History

This section logs any updates or changes made to the document.

09/01/2025 – The start of creating this document.

Appendix

Reference:

A P, Shanmugam & A, Ramalakshmi & Ganeshan, Sasthri & S, Baalachandran. (2020). Library Management System. Xi'an Jianshu Keji Daxue Xuebao/Journal of Xi'an University of Architecture & Technology. 12. 743-753. 10.37896/JXAT12.11/29777.