**Library Management System Using Data Structures in C++**

**Project Description:**

The proposed project is a comprehensive Library Management System implemented in C++ to efficiently manage books, checkouts, returns, and requests. The project leverages multiple data structures to handle different operations, making it an ideal educational tool to showcase their practical applications. This system demonstrates the use of arrays, linked lists, doubly linked lists, stacks, queues, and binary search trees (BSTs) to solve common library management problems.

**Objectives:**

1. To create a functional and efficient library management system.
2. To utilize various data structures for specialized tasks:

* Arrays for fixed book storage.
* Linked Lists for dynamic book storage and management.
* Doubly Linked Lists for maintaining checkout and return history.
* Stacks for handling recent checkouts (LIFO functionality).
* Queues for managing book requests or waiting lists (FIFO functionality).
* BSTs for organizing and searching books by title or author.

3. To enhance understanding of data structures and their applications in real-world scenarios.

**Features:**

**1.Array Implementation:**

Store a fixed list of book IDs and titles for static library records.

**2.Linked List:**

Add and display dynamically added books.

**3.Doubly Linked List:**

Manage and traverse book checkout and return history in both forward and reverse order.

**4.Stack:**

Track recent book checkouts with "last-in, first-out" functionality.

**5.Queue:**

Maintain a list of book requests in the order they are received.

**6.Binary Search Tree (BST):**

Organize books by title or author and enable efficient searching.

**Methodology:**

**Data Structures:**

* Design and implement each data structure to handle a specific aspect of library operations.

Integrate all components into a unified system.

**Programming Language:**

The project will be written in C++ to take advantage of object-oriented programming (OOP) features such as classes and structures.

**Development Phases:**

**Phase 1:** Define the structure of books and implement basic operations using arrays.

**Phase 2:** Develop and test dynamic book management with linked lists and doubly linked lists.

**Phase 3:** Implement stack and queue functionalities for book checkouts and requests.

**Phase 4:** Integrate a binary search tree (BST) for efficient book searching.

**Phase 5**: Finalize the system by connecting all components and adding user interaction.

**Expected Outcomes:**

**1.**A fully functional Library Management System that handles book storage, checkouts, returns, and search operations efficiently.

**2.**Practical demonstration of how different data structures solve specific problems in library management.

**3.**A reusable codebase for educational purposes, illustrating the real-world applications of arrays, linked lists, stacks, queues, and BSTs.

**Tools and Technologies:**

**Language: C++**

**IDE: Dev C++;**

**Operating System: Windows.**

**Libraries: Standard Template Library (STL) for stack and queue**