Library Management System

SUBMITTED BY

EMAZ ALI KHAN (65566)

ABDUL RAFAY (65540)

WALI AJAZ UL HAQ (65537)

SUMAIKA ASIF (65651)

SUPERVISED BY

DR. NASIR TOUHEED



PROJECT REPORT SUBMITTED TO THE FACULTY OF COMPUTING, KARACHI INSTITUTE OF ECONOMICS AND TECHNOLOGY, IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE/ SOFTWARE ENGINEERING

SPRING 2024



**PROJECT CERTIFICATE**

*This is to certify that the Project titled, “Library Management System”, is submitted to the Department of Computer Science, Spring 2023, by Emaz Ali Khan (*65566*), Abdul Rafay (65540), Wali Ajaz Ul Haq (65537) and Sumaika Asif (65651) for the award of the degree of Bachelor of Science in the discipline of Computer Science/Software Engineering. The Project has been carried out under my supervision. I certify that the work submitted is original and not plagiarized from any other source, except as specified in the references. Neither the Project nor the work contained therein has been previously submitted to any other institution for a degree.*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Faculty name and signature

**KARACHI INSTITUTE OF ECONOMICS AND TECHNOLOGY**

# ORIGINAL LITERARY WORK DECLARATION

|  |  |  |
| --- | --- | --- |
| **PROGRAM** | *BS-CS* | |
|  |  | |
| Student Name: Emaz Ali Khan, Abdul Rafay, Wali Ajaz Ul Haq, Sumaika Asif | | Reg. No: 65566 , 65540 , 65537 , 65651 |
| Email: --- | | Mobile No: --- |

Project Title:

|  |
| --- |
| “Library Management System”  We do solemnly and sincerely declare that:  1. We are the author of this work;  2. This work is original;  3. Use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work; I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;  4. We hereby assign all and every rights in the copyright to this work to Karachi Institute of Economics and Technology (KIET), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of KIET having been first had and obtained;  5. We are fully aware that if in the course of making this work I have infringed any copyright whether intentionally or otherwise, We may be subject to legal action or any other action as may be determined by KIET. |

Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Place: Karachi-Pakistan.

**Acknowledgments**

# بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ

# We would like to express our special gratitude to our reputable instructor, Dr. Nasir Touheed, who provided us with the study materials, and helped us understand a lot of Data Structure & Algorithms concepts. We put in a lot of effort to finish this Library Management System project. However, without our instructor's guidance, we would not have been able to complete this project. We are extremely thankful to our instructors for giving us the opportunity to work on a new project.

**Executive Summary**

The DSA Project, focused on implementing a Library Management System, utilizes a diverse set of data structures and algorithms to efficiently manage books and ebooks within a library context. The system employs a modular approach, dividing functionalities into distinct classes and methods for book and ebook management.

**Code Explanation:** The project encompasses several key components:

* Book and Ebook Classes: These classes serve as data holders for book and ebook information, encapsulating attributes and providing a foundation for further operations.
* LinkedList and Stack: These data structures are employed to manage collections of Book and Ebook objects, respectively. They offer functionalities such as appending, removing, and searching for items based on specific criteria.
* Queue: Utilized specifically for managing available Book objects, the queue structure facilitates operations like enqueuing and dequeuing.
* Library Management System Class: Serves as the orchestrator of library operations, encompassing methods for managing books and ebooks, interacting with users, and handling file operations.

**Data Structures and Operations:** The project employs various data structures, including linked lists, stacks, queues, and arrays (Python lists), each serving a specific purpose in managing library resources efficiently.

* Linked List: Facilitates operations on collections of Book objects, allowing for efficient appending, removing, and searching based on title, author, or ISBN.
* Stack: Manages collections of Ebook objects, following the Last-In-First-Out (LIFO) principle and enabling operations like pushing, popping, and searching.
* Queue: Specifically used for managing available Book objects, adhering to the First-In-First-Out (FIFO) principle and supporting operations like enqueuing, dequeuing, and checking emptiness.
* Array (Python List): Maintains a list of available Ebook objects, providing fast access and supporting iteration, appending, removing, searching, and file operations.
* Searching: The linked list is searched for book details that match the search query (title, author, or ISBN), while the stack is checked for ebook details matching the search query (title or author).

Library Management System

Emaz Ali Khan (65566)

Abdul Rafay (65540)

Wali Ajaz Ul Haq (65537)

Sumaika Asif (65651)

Karachi Institute of Economics and Technology

Author Note

Department of COCIS, KIET City Campus (North Nazimabad): F-103 & 103/1, Block-F, Allama Rasheed Turabi Road, Near Ziauddin Roundabout, Karachi, 75400, Sindh, Pakistan.

# CHAPTER 1

# INTRODUCTION

* 1. **ABOUT PROGRAM**

This program simulates a library management system using various data structures to efficiently manage books and ebooks.

**Key Functionalities:**

* **Book Management:**
  + Represents books using the Book class with attributes like title, author, ISBN, and availability status.
  + Implements functionalities like:
    - Adding new books to the system.
    - Removing books based on specific criteria.
    - Searching for books by title, author, or ISBN.
    - Viewing a list of available books.
  + Utilizes a LinkedList data structure to manage the book collection, enabling efficient search and retrieval.
  + Maintains a separate Queue to track available books for borrowing.
* **Ebook Management:**
  + Represents ebooks using the Ebook class with attributes like title, author, and format.
  + Implements functionalities like:
    - Adding new ebooks to the system.
    - Removing ebooks based on specific criteria.
    - Searching for ebooks by title or author.
    - Viewing a list of available ebooks.
  + Utilizes a Stack data structure to manage the ebook collection, following the Last-In-First-Out (LIFO) principle.
  + Maintains a separate list to track available ebooks for access.
* **Data Persistence:**
  + Loads book and ebook data from external files at startup.
  + Saves any changes made to the book and ebook collections back to their respective files.

**Data Structures:**

* **LinkedList:** Efficiently manages the book collection, allowing for easy search and retrieval by title, author, or ISBN.
* **Stack:** Manages the ebook collection, following the LIFO principle for adding and removing ebooks.
* **Queue:** Tracks available books for borrowing, following the First-In-First-Out (FIFO) principle.
* **Array (Python List):** Stores available ebooks, enabling fast access and search operations.

**Searching:**

* **Books:** The linked list is traversed to compare book details with the search query (title, author, or ISBN) for a match.
* **Ebooks:** The stack is iterated through, comparing ebook details with the search query (title or author) for a match.

Overall, this program provides a comprehensive library management system that leverages various data structures for efficient book and ebook management, search functionalities, and data persistence.

* 1. **OBJECTIVE**

To design and explain a Library Management System that efficiently manages both physical books and ebooks using appropriate data structures and algorithms. This system aims to:

* **Streamline book and ebook management:** Facilitate adding, removing, searching, and viewing available resources.
* **Maintain accurate data:** Track book and ebook information like title, author, format, and availability status.
* **Enable user interaction:** Allow users to receive and return books, manage ebooks, and view available options.
* **Utilize efficient data structures:** Implement linked lists for books, stacks for ebooks, and queues for available books, ensuring optimal search and management operations.
* **Facilitate data persistence:** Load and save book and ebook data from files for long-term storage and retrieval.

# CHAPTER 2

## FEATURES

* **Book Management:**
  + Add new books with title, author, ISBN, and availability status.
  + Receive and return books, updating their availability status and saving transactions.
  + Search for books by title, author, or ISBN.
  + View a list of available books.
  + Remove books based on specified criteria (title, author, ISBN) and update file.
* **Ebook Management:**
  + Add new ebooks with title, author, and format.
  + Search for ebooks by title or author.
  + View a list of available ebooks.
  + Remove ebooks based on specified criteria (title, author, format) and update file.
* **Data Structures:**
  + Utilizes linked lists for efficient book management (adding, removing, searching).
  + Utilizes stacks for managing ebook collections (LIFO principle).
  + Utilizes queues to manage available books (FIFO principle).
  + Employs an array (Python list) for storing and accessing available ebooks.
* **Search Functionality:**
  + Supports searching books and ebooks based on title, author, and ISBN/format.
  + Search algorithms efficiently traverse the chosen data structure (linked list/stack).
* **File Management:**
  + Loads book and ebook data from separate files upon startup.
  + Saves updated book and ebook information back to their respective files.
* **Menu-Driven Interface:**
  + Provides a user-friendly menu for interaction and selection of desired operations.
  + Guides the user through various functionalities of the library management system.

These features enable the system to effectively manage book and ebook collections, facilitate user interaction, and maintain accurate data through file operations.

**CHAPTER 3**

**ADVANCEMENT**

This Library Management System goes beyond traditional methods by utilizing advanced data structures like linked lists for efficient book searches, stacks for managing the latest ebooks, and queues for fair lending practices regarding available books. It also leverages a separate array for quick access to available ebooks. This design streamlines operations, automates tasks, and provides a user-friendly interface for searching, managing, and interacting with library resources. File integration ensures data persistence and facilitates restoration when needed. Overall, these advancements contribute to a more efficient, organized, and accessible library experience compared to older systems.

**CHAPTER 4**

**CONCLUSION**

In conclusion, this Library Management System represents a significant advancement over traditional methods. By leveraging data structures like linked lists, stacks, and queues, it efficiently manages both physical books and digital ebooks, enabling swift searching, adding, removing, and tracking of resources. The system also boasts a user-friendly menu interface and seamless file integration for data persistence. These advancements contribute to a more streamlined and organized library experience, improving efficiency, accessibility, and overall user satisfaction. The system offers a robust foundation for further development, potentially incorporating additional features such as user accounts, reservation systems, and advanced reporting capabilities, further enhancing the library's functionalities and services.