

**PURBANCHAL UNIVERSITY INSTITUTE OF SCIENCE**

**AND TECHNOLOGY**

**KIST COLLEGE OF INFORMATION TECHNOLOGY**

## A Project Proposal On

**Movie Ticket Booking System**

**Submitted By:**

**Sushan Bhandari, Shakshyam Regmi, Sagar Tamang**

**Submitted To:**

Department of Science & Technology

BIT

Kamalpokhari, Kathmandu

16 June, 2025

# ACKNOWLEDGEMENT

First of all, we would like to express our deep sense of gratitude to our head of department **Mr. Dipak Khakda**, Department of Bachelor in Information, KIST College of Science and Information Technology for providing us with such an opportunity to prepare a report on " **Library Management System**”. We would like to thanks all the teaching and non - teaching faculties’ members of the **Kist College** for their continuous support, guidance, unparalleled suggestions, and encouraging opinions throughout the completion of the project. We would also like to express our special thanks of gratitude to our friends and family who helped us and have been a constant source of inspiration during the preparation of this report.

Group members:

-Sagar Tamang

-Susan Bhandari

-Shakshyam Regmi

# ABSTRACT

This proposal presents the development of a specialized system designed to display information of movies available in a theatre, including their time and date for efficient ticket booking. This system aims to make ticket booking for movies easier and more efficient by providing a user-friendly interface. It focuses on showing movie details, booking tickets and managing the data of customers. It also shows the seating pattern which further helps the customers to book the seat they require.

*Keyword:*

**TABLE OF CONTENTS**

[ACKNOWLEDGEMENT i](#_Toc6100)

[ABSTRACT ii](#_Toc6101)

## Chapter 1: INTRODUCTION .................................................................................... 1

1.1 Background .............................................................................................................. 1

1.2 Motivation ................................................................................................................ 1

1.3 Problem Definition ................................................................................................... 2

1.4 Objectives ................................................................................................................ 2

1.5 Scope and Applications ............................................................................................ 2

1.6 Requirements ........................................................................................................... 2

**Chapter 2: LITERATURE REVIEW ........................................................................ 3**

**Chapter 3: METHODOLOGY ................................................................................... 4**

3.1 Algorithm ................................................................................................................. 4

**Chapter 4: EXPECTED OUTCOME ........................................................................ 5**

## Reference ...................................................................................................................... 6

## Chapter 1: INTRODUCTION

Library Management System is software developed in the C programming language that enables efficient handling of common library tasks such as adding and removing books, managing members, issuing and returning books, and viewing available records. The system simplifies the work of librarians and makes book and member data accessible and manageable through a menu-based console application. It also implements file handling to store data persistently

1.1 Background

Despite the increasing use of digital resources, physical libraries are still widely used, especially in educational institutions. Managing these libraries manually can be time-consuming and error-prone. With limited access to large commercial library systems, many schools and colleges benefit from simpler, tailor-made solutions. This system was created to address the need for a compact and efficient library management tool, using C to demonstrate fundamental concepts in structured programming and file management.

## 1.2 Motivation

The motivation behind this project stems from the desire to digitize and simplify the library management process. Manual systems often lead to misplaced records, inefficiency in book tracking, and excessive time consumption. This project also serves as an academic challenge—to apply theoretical programming knowledge in a practical, real-life application that can be used in educational or small institutional libraries.

## 1.3 Problem Definition

Traditional library systems based on pen-paper methods are prone to data loss, mismanagement, and delays. Finding books or member information quickly becomes difficult, and there is no track of issued or returned books. Thus, a system is required that offers basic functionality for managing books and users, while maintaining records safely using file storage mechanisms.

## 1.4 Objectives

The main objectives of our project are listed below**:**

* To design and implement a menu-driven Library Management System using C.
* To manage books and members effectively.
* To issue and return book updates accordingly.
* To use file handling to ensure data presistance.

## 1.5 Scope and Applications

This project applies to schools, colleges, or small institutions where libraries are maintained manually. The system is limited to basic functionalities such as book and user management, issuing, returning, and viewing records. It does not include features like online access or mobile support but lays the foundation for future enhancements.

**1.6 Requirements** Hardware:

1. Computer
2. CPU
3. Keyboard
4. Mouse Software:
5. DEV C++/ Code Blocks
6. Windows 10
7. MS Word or WPS Office

## Chapter 2: LITERATURE REVIEW

Several studies and project implementations have shown the use of basic programming languages to manage library-related operations. Many projects created in C or C++ focus on console-based applications, as they offer a direct way to understand memory management, file handling, and structured logic. For instance, basic C-based systems often use structures for book data and binary files forpersistence.  
  
In another similar project, books were added and retrieved using arrays, but this lacked data persistence. Our project advances by including file operations that allow records to be retained even after program termination. Furthermore, the simplicity and accessibility of C make it a widely chosen language for such foundational software projects.

## Chapter 3: METHODOLOGY

The Library Management System is designed with simplicity and usability in mind. It follows a menu-driven architecture where users interact through keyboard inputs. The system uses C language structures to store book and member information, and file handling to preserve data between program executions. Each functionality such as adding books, deleting records, or issuing a book is implemented as a separate function to promote modular design.

## 3.1 Algorithm

Step 1: Start

Step 2: Display login and signup options

Step 3: If selected

Login: go to step 4

Signup: go to step 5

Step 4: Input Details (name, number and password), save them and return to step 2

Step 5: Input number and password

Step 6: If

Correct: go to step 7

Incorrect: go to step 5

Step 7: Display Menu (Add book, remove book, add members, etc) as numbers

Step 8: if

**Add Flowchart Compulsory**

## Chapter 4: EXPECTED OUTCOME

Upon execution of the project, the Library Management System will provide a functional interface for managing books and library users through a terminal-based environment. It will support the addition, deletion, search, issue, and return of books along with storage of data using file handling. The outcomes include:

# Reference

F. Ros,"Simple Library Management System In C With Source Code," 2023.  
  
S. L. Malavikka R, "Library Management Project in C," 2024. [Online].  
  
GeeksforGeeks, "C File Handling and Structure," 2022. [Online].

1