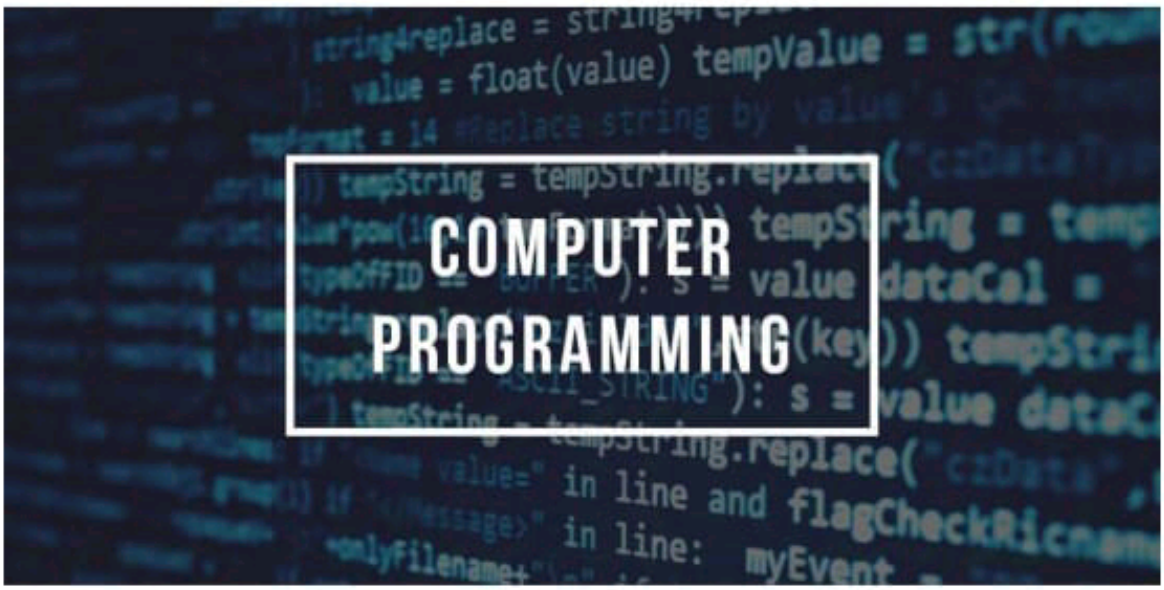
CHANDIGARH UNIVERSITY

Discover. Learn. Empower.



Case study: E-library management system

Submitted by:

Manjot kaur(UID:24BCA10110)

Navpreet kaur(UID:24BCA10107)

Gursimran kaur(UID:24BCA10112)

Class-24BCA-2(B)

Subject code-24CAH-101

Submitted to Mr.ArvinderSingh

CERTIFICATE

This is to certify that Manjot kaur, Navpreet Kaur, Gursimran kaur Uid-24BCA10110, 24BCA20107, 24BCA10112, Bachelor of computer applications 1'st Semester students of CHANDIGARH UNIVERSITY has done Project work tittle E-library management system. Under the guidance of our senior faculties towards the fulfillment of awards of Bachelor of computer applications during the period of OCTOBER 2024 to NOVEMBER 2024.

Date: Signature

ACKNOWLEDGEMENT

We would like to convey my heart felt gratitude to Mr. Arvinder Singh For his tremendous support and assistance in the completion of my project. We would also like to thank our faculty, for providing us with this wonderful opportunity to work on a project with the topic-E-library management system. The completion of the project would not have been possible with out their help and in sights.

Manjot Kaur

Gursimran Kaur

Navpreet Kaur

► INDEX

C programming: C is a high-level, general-purpose programming language developed in the early 1970s by Dennis Ritchieat BelLabs. It's one of the most widely used programming languages, especially for systems programming, because of its efficiency and control over hardware. Here's a breakdown of what C is and why it remains popular.

▶ What is C?

General Purpose: C is designed to be versatile, a lowing developers to create a variety of applications, from operating systems to applications.

Compiled Language: C code is converted directly in to machine code, which make sit fast and efficient compared to interpreted languages.

Low-Level Features: C provides access to low-level memory management (like pointers) and bitwise operations, which gives more control over the hardware.

Structured Language: C allows programs to be broken into smaler, manageable functions, making code easier to read, understand, and debug.

▸Why Use C?

-Performance: C is highly efficient because it's close to assembly language. It's still widely used in applications where performance is critical, such asembelled systems, operating systems, and real-time systems.

Portability: C code is portable across different hardware architectures with minimal modification, which makes it suitable forwriting system software that needs to run on various machines.

Foundation for Other Languages: Many modern programming languages(like C++,C, java and even Python)are influenced by C. Learning C helps understand how these languages work at a lower level.

Control and Flexibility: C allows direct manipulation of hardware and system resources through pointers and manual memory management, which offers a higher level of control than many modern languages.

Small Language Core: C has are relatively small set of key words and a simple structure, making it easier to learn and focus on fundamental programming concepts.

▸The Need for C-Systems Programming:

Operating systems, embedded systems, and device drivers require direct access to hardware, and C's close-to-hardware capabilities make it ideal for these purposes.

Resource-Constrained Environments : C is light weight and doesn't require much overhead, making it suitable for systems with limited resources.

Reliability and Stability: Many legacy systems are written in C and continue to run reliably, which sustains its demand in maintaining, upgrading, and extending existing systems.

Learning Foundation: For new programmers, C provides an excellent foundation in fundamental programming concepts, data structures, and algorithm development, which are essential for understanding other languages and advanced programming topics.

E-Library Management Systems

▶Introduction to E-Library Management System

The E-Library Management System (ELMS) is a digital platform designed to facilitate the management and accessibility of library resources in a virtual environment. With the rapid advancement of technology and the increasing need for efficient information retrieval, traditional library systems are evolving in to electronic formats that provide a range of features aimed at improving user experience and operational efficiency.

▶Purpose of the E-Library Management System

The primary goal of an E-Library Management System is to streamline library operations, making it easier for both librarians and users to access and manage books, journals, and other educational resources. By transitioning from physical to digital systems.

E-LMS aims to:

Enhance Accessibility: User scan access library resources from anywhere and at any time, breaking down geographical barriers to information.

Improve Efficiency: Auto mating tasks such as cataloging, tracking, and lending allows librarians to focus on more critical aspects of library management.

Facilitate Information Retrieval: Advanced search functions enable users to find resources quickly and easily, using various criteria such as title, author, orgenre.

Support Digital Resources: With the increasing availability of e-books and online journals, E-LMS accommodates diverse formats, catering to different learning styles and preferences.

Manage User Data: The system keeps track of user registrations, borrowing history, and fines, enhancing the overall management of library patron data.

▶Key Features of an E-Library Management System User Registration:

Allows new users (students, faculty, etc.) to create accounts and manage their profiles.

Book Management: Enables librarians to add, update, and delete book records, including physical and digital resources.

Search and Browse: offers a user-friendly interface for searching books and other resources by various parameters.

Issuing and Returning Books: Facilitates the process of lending books to users and tracking their return, including due dates and fines for late returns.

Digital Resource Management: Supports the inclusion of e-books, audio books ,and other digital formats along side traditional print materials.

Reports and Analytics: Provides in sights in to library usage, popular resources, and circulation statistics, aiding in informed decision-making.

Security and Privacy: Ensures user data protection through secure log in mechanisms and data encryption.

►ImportanceofE-Library ManagementSyste ms

Intoday'sinformationage,wheredigitalliteracyisparamount,E-Library ManagementSystemsplayacriticalroleineducationalinstitutions,publiclibraries ,and researchcenters.Theynotonlysimplifythemanagementofvastamountsof informationbutalsopromotelifelonglearningbymakingresourcesreadilyavailabl eto

Scope:

The system is ideal for educational institutions, public libraries, or private book collections, providing librarians and users with a centralized, computerized platform. It offers essential features like book cataloging, user registration, book issue and return, search functionality, and inventory management.

In this article, we will discuss the approach to creating an E-Library Management System where the user has the following options:

- Add book information.
- Display book information.
- To list all books of a given author.
- To list the count of books in the library.

Functionalities Required:

- If the user tries to add a book then the user must have to provide the below specific Information about the book:
    - Enter Book Name:
    - Enter Author Name:
    - Enter Pages:
    - Enter Price:

- When the user tries to display all books of a particular author then the user must have to enter the name of the author:
    - Enter the author's name:
- The E-Library Management System must be also capable of counting all the books available in the library.

Advantages of E-Library Management System

24/7 Accessibility: Users can access resources at any time from anywhere with an internet connection, allowing for greater flexibility and convenience, especially for remote or online learners.

Efficient Resource Management: ELMS allows libraries to keep accurate, up-to-date records of books and materials, track inventory, and manage borrowing and returns efficiently. Automated cataloging and record-keeping reduce manual effort and minimize errors.

Reduced Operational Costs: By reducing the need for physical paperwork, manual record-keeping, and in-person library management tasks, ELMS helps libraries save on labor and material costs over time.

Enhanced User Experience: ELMS offers search and filtering options for books by title, author, genre, and other criteria, making it easier for users to find what they're looking for quickly. Personalized accounts with features like notifications, recommendations, and borrowing history improve user engagement.

Environmental Benefits: Moving to digital records and digital resources like e-books reduces the need for paper and physical copies, supporting environmentally sustainable practices.

Data Analytics and Reporting: ELMS provides data on user behavior, popular books, and circulation trends, which can help libraries make informed decisions on resource allocation and acquisitions.

Enhanced Security and Data Backup: User data is securely stored with ELMS, and regular backups protect information from being lost due to system failures or other disruptions.

- **Disadvantages of E-Library Management System**

  Initial Setup Costs and Maintenance: Implementing an ELMS can be costly due to the need for software, hardware, and possibly training staff. Regular maintenance, software updates, and technical support may require additional resources.

  Dependence on Technology and Internet: ELMS is entirely reliant on technology and internet connectivity. In areas with poor internet infrastructure, this can hinder accessibility. Downtime due to power outages, system failures, or server issues can disrupt library operations.

  Cybersecurity Risks: Storing sensitive user data online can expose the system to cyber threats, including data breaches and hacking, which could compromise user privacy. Libraries must invest in robust cybersecurity measures to safeguard user data.

  Lack of Personal Interaction: An ELMS may reduce face-to-face interactions between librarians and users, potentially affecting user engagement and support, especially for those who need help navigating library resources.

  Learning Curve for Staff and Users: Staff and users unfamiliar with digital systems may face challenges adapting to ELMS, requiring time and training to become proficient. Users who prefer traditional methods may initially resist the digital transition, impacting overall user satisfaction

  Data Loss and Technical Issues: If not regularly backed up, a system crash or technical failure could lead to data loss. Technical problems, such as software bugs or compatibility issues, may interrupt library services until resolved.

  Digital Divide: Users who lack access to computers, smartphones, or reliable internet, such as those in rural areas, may be disadvantaged when libraries transition fully to an electronic system.

► Below is the program to implement the E-Library Management System:

```
// C program for the E-library
// Management System
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

// Create Structure of Library
Struct library {
   Char book_name[20];
   Char author[20];
   Int pages;
   Float price;
};

// Driver Code
Int main()
{
   // Create a instance
   Struct library lib[100];

   Char ar_nm[30], bk_nm[30];
```

```
//Keep the track of the number of
//of books available in the library
Int I, input, count;

I = input = count = 0;

//Iterate the loop
While (input!=5) {

   Printf( "\n\n*******######"
         "WELCOME TO E-LIBRARY "
         "#####********\n" );
   Printf( "\n\n1. Add book infor"
         "mation\n2. Display "
         "book information\n" );
   Printf( "3. List all books of "
         "given author\n" );
   Printf(
      "4. List the count of book"
      "s in the library\n" );
   Printf( "5. Exit" );

   //Enter the book details
   Printf( "\n\nEnter one of "
```

```
        "the above: ");
Scanf("%d",&input);

//Process the input
Switch (input) {

//Add book
Case 1:

   Printf("Enter book name= ");
   Scanf("%s",lib[i].book_name);

   Printf("Enter author name= ");
   Scanf("%s",lib[i].author);

   Printf("Enter pages= ");
   Scanf("%d",&lib[i].pages);

   Printf("Enter price= ");
   Scanf("%f",&lib[i].price);
   Count++;

   Break;
```

```
//Printbookinformation
Case2:
   Printf("youhaveentered"
         "thefollowing "
         "information\n");
   For(I=0;I<count;i++) {

      Printf("booknname=%s",
            Lib[i].book_name);

      Printf("\tauthornname=%s",
            Lib[i].author);

      Printf("\t pages=%d",
            Lib[i].pages);

      Printf("\t price=%f",
            Lib[i].price);
   }
   Break;

//Takeheauthornameasinput
Case3:
   Printf("Enterauthornname:");
```

```
        Scanf( "%s" , ar_nm);
        For (I= 0; I < count; i++) {

          If (strcmp(ar_nm,
                Lib[i].author)
            == 0)
            Printf( "%s %s %d %f" ,
                Lib[i].book_name,
                Lib[i].author,
                Lib[i].pages,
                Lib[i].price);
        }
        Break;

    //Print total count
    Case 4:
        Printf( "\n No of books in "
            "brary : %d" ,
            Count);
        Break;
    Case 5:
        Exit(0);
    }
}
```

```
        Scanf( "%s" , ar_nm);
        For (I= 0; I < count; i++) {

          If (strcmp(ar_nm,
                Lib[i].author)
            == 0)
            Printf( "%s %s %d %f" ,
                Lib[i].book_name,
                Lib[i].author,
                Lib[i].pages,
                Lib[i].price);
        }
        Break;

    //Print total count
    Case 4:
        Printf( "\n No of books in "
            "brary : %d" ,
            Count);
        Break;
    Case 5:
        Exit(0);
    }
}
```

```
    Return 0;
}
```

- **Output**

  Displaying the functionalities and input

  For option 1:

```
*********######WELCOME TO E-LIBRARY #####
     ********


1. Add book information
2. Display book information
3. List all books of given author
4. List the count of books in the library
5. Exit

Enter one of the above: 1
Enter book name = little woman
Enter author name =lauisa may alcott.
    Enter pages = 1350
Enter price = 450
```

For Option 2:

```
*********######WELCOME TO E-LIBRARY #####
    ********


1. Add book information
2. Display book information
3. List all books of given author
4. List the count of books in the library
5. Exit

Enter one of the above:2.
    Enter book name = a christmas carol
    Enter author name = charles d Dickens
    Enter pages = 1470.
    Enter price = 570
```

Foroption3:

```
*********######WELCOME TO E-LIBRARY #####
    ********


1. Add book information
2. Display book information
3. List all books of given author
4. List the count of books in the library
5. Exit

Enter one of the above:3.
    Enter book name = treasure Island.
    Enter author name = robe Robert Louis
    Stevenson
    Enter pages = 1290.
    Enter price = 750
```

For option 4:

```
*********######WELCOME TO E-LIBRARY #####
    ********


1. Add book information
2. Display book information
3. List all books of given author
4. List the count of books in the library
5. Exit

Enter one of the above:4
    Enter book name = Moby dick.
    Enter author name =Herman Melville.
    Enter pages = 1460
=== Session Ended. Please Run the code again
    ===
```

Conclusion

In conclusion, this E-Library Management System in C programming provides a basic framework for managing books within a library. The program demonstrates essential functionalities, including adding, viewing, searching, and deleting books using a structured approach. It employs fundamental concepts like structures, arrays, and menu-driven loops, making it a practical exercise for learning C programming and data management.

Though limited in its current form, this system could be further expanded to include more sophisticated features, such as persistent storage (saving to and reading from files), sorting options, or enhanced search functionalities. Overall, this project serves as an excellent starting point for developing more complex library or database management applications in C.