Department of Computer Science & Engineering

Project Report

Project Title: Library Management System

Course Title: Software Development Lab I

Course Code: CSE-136

Batch No: 31st

Semester: 3rd Semester

Section: E

Team: 1

Submitted by	Submitted to
Name: Md Borhan Uddin Ashik ID: 223311161	A.S.M. Delwar Hossain Lecturer Department Of CSE Varendra University
Name: Abu Raihan Md Sohag ID: 223311162	
Name: Afia Akhter ID: 223311163	_____ *Signature*

Date of Submission: 28/12/2023

Motivation:

After seeing Varendra University library system, I feel that it needs to be more student friendly and improved. Only the librarian could know whether there are any required books in the library, the books of any publication, how many books are in stock through internal apps. Which seems like a lot of trouble and time consuming for students like us. We thought how to improve this system and bring that service to the student level.

Objectives:

The primary objectives of this project are to:

1. Develop a student-friendly and efficient LMS: The system will provide a seamless interface for librarians and patrons, enabling them to manage library resources and access information with ease.
2. Enhance library operations: Streamline book cataloging, circulation management, allowing librarians to focus on providing exceptional service to patrons.
3. Every student can easily find his desired book without going to the library or asking the librarian.
4. Student can also see here his total number of books issued, total number of publications and number of categories of books.
5. Librarians can view all issues and easily view stock out books.
6. Librarians can easily add new books to the library.

Overview:

♦ **Description of the Project:**

The Library Management System is a software application developed in C++. It is designed to overcome the manual record-keeping process in libraries.
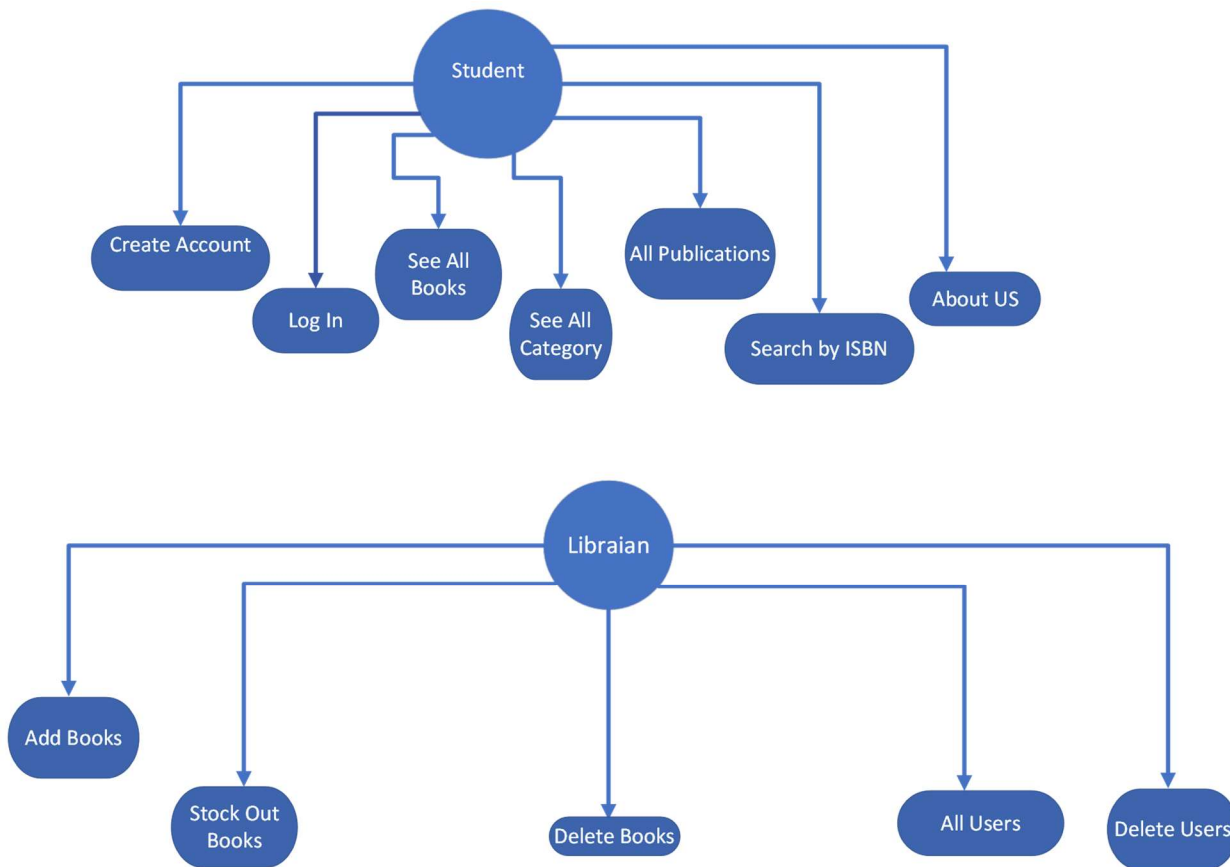
- The scope of a Library Management System in C++ includes but is not limited to:
- Keeping track of the return dates and checking for overdue books.
- Providing a user-friendly interface for librarians to manage the system.
- Develop a comprehensive cataloging system for efficient organization and retrieval of library resources.
- Enable users to search for books based on various criteria, including book name etc.
- Create and manage user profiles, storing relevant information such as borrowing history and personal details.

♦ **Materials and Methods:**

The project is developed using the C++ programming language, leveraging its Object-Oriented Programming (OOP) capabilities. The system is designed around various classes representing entities such as 'LMS', 'Users', and 'Books'. These classes encapsulate the data and provide methods for interacting with that data, adhering to the principles of encapsulation, inheritance, and polymorphism.

As for the Data Flow Diagram (DFD), it would typically include entities like 'Librarian', 'System', and 'Database', and show how data flows between them during different processes like 'Add/Update/Delete Book', 'Issue Book', 'Return Book', etc. Unfortunately, I can't draw a DFD here, but there are many online tools available that you can use to create one.

Use case Diagram:

Features:

1. **User-Friendly Interface:** The system has a simple and intuitive interface, making it easy for librarians and users to interact with the system.
2. **Efficient Data Structures:** The system uses efficient data structures provided by the Standard Template Library (STL) in C++, ensuring fast retrieval and storage of data.
3. **Comprehensive Record Keeping:** The system keeps track of various details such as books issued, due dates, late fees, etc. This helps in maintaining a clear record of all transactions.
4. **Search Functionality:** The system provides a search functionality for book and member records. This helps in quickly finding the required information.
5. **Data Consistency and Integrity:** The system ensures data consistency and integrity. Any changes in the data are accurately reflected across the system.

Timeline:

The project is expected to take 2 months to complete. The following is a tentative timeline:

Phase 1: Project Planning and Design (November)

- Requirements gathering
- System design and architecture
- User interface prototyping

Phase 2: Development (December)

- Core functionality development
- Unit testing and integration testing
- User acceptance testing

Software Requirements/Tools:

To successfully complete the Library Management System project in C++, you would need the following software tools:

1. **C++ Compiler:** A C++ compiler is required to compile and run your C++ code. Examples include GCC, Clang, and MSVC.
2. **Integrated Development Environment (IDE):** An IDE can help you write, compile, and debug your code more efficiently. Examples include Code Blocks and Visual Studio.
3. **Version Control System:** A version control system like Git can help you manage different versions of your project, especially if you are working in a team.
4. **UML Tool:** A tool for creating UML diagrams (like Data Flow Diagrams) can be useful for planning and visualizing your project. Examples include Draw.io, Lucid chart, and Microsoft Visio.
5. **Text Editor:** A text editor can be useful for writing and editing your project documentation. Examples include Visual Studio Code.

Functionality with code snips & output:

Main Interface

```cpp
// main function
void mainFunc(LMS &ms, function<void(LMS lms)> LogedUserMenu)
{
    int menu;
    do
    {
        cout << MAGENTA << "\n\n*******************\nWelcome Big & Not Found Library :)\n*******************\n\n"
             << RESET << "1.SignUp\n2.LogIn\n3.Show All Books\n4.Show All Category\n5." << CYAN << "About Us\n"
             << RESET
             << "6." << RED << "Exit\n\n";
        cout << RESET << "Enter your choice: ";
        cin >> menu;
        switch (menu)
        {
        case 1:
        {
            lms.signup();
        }
        case 2:
        {
            map<int, User> users = loadUsers();
            pair<int, string> loginData = logIn();
            if (users.count(loginData.first) && users[loginData.first].password == loginData.second)
            {
                isUserLoged = true;
            }
            else if (loginData.first == 61 && loginData.second == "admin")
            {
                lms.adminMenu();
            }
            else
            {
                cout << RED << "\nLogin failed. Invalid ID or password.\nlogin again\n"
                     << RESET;
            }
            if (isUserLoged)
            {
                LogedUserMenu(lms);
            }
            break;
        }
        case 3:
        {
            lms.loadBooks();
            lms.showBooks();
            break;
        }
        case 4:
        {
            cout << "\nall books in our Category: \n";
            lms.showCategory();
            break;
        }
        case 5:
        {
            lms.aboutUs();
        }
        case 6:
        {
            exitProgram();
            break;
        }

        default:
            cout << RED << "invalid choice! please choose a valid choice :)\n"
                 << RESET;
        }
    } while (menu != 6);
}
```

Code

Output

```
*******************
Welcome Big & Not Found Library :)
*******************

1.SignUp
2.LogIn
3.Show All Books
4.Show All Category
5.About Us
6.Exit

Enter your choice: █
```

5

1.Signup

```cpp
// sign up function
void signup()
{
    cout << "Fill the form to get library card :) \n";
    cout << "Name: ";
    string name;
    cin.ignore();
    getline(cin, name);
    int id;
    cout << "id: ";
    cin >> id;
    cout << "email: ";
    string email;
    cin.ignore();
    getline(cin, email);
    char ch = ' ';
    string password;
    cout << "You don't have permit to backspace\nPassword: ";
    while (ch != 13)
    {
        ch = getch();
        if (ch == 8 || ch == 13 || ch == 32)
            continue;
        else
        {
            password += ch;
            cout << "*";
        }
    }
    // admin
    if (id == 223311161 && password == "admin")
    {
        User adminUser(name, id, email, password);
        saveUser(adminUser);
        cout << "\nCongrats! Admin user added successfully.\n";
    }
    // normal people
    else
    {
        // Create a new user
        User newUser(name, id, email, password);

        saveUser(newUser);
        cout << "\nCongrats! You have successfully created an account and got a library card :)\n";
    }
}
```

```
password("") {}
ring &e, const string &p)
ord(p) {}
```

Code

Output

```
Enter your choice: 1
Fill the form to get library card :)
Name: Saladin Aiyubi
id: 1187
email: aiyubi@mail.com
You don't have permit to backspace
Password: ****
Congrats! You have successfully created an account and got a library card :)
Fill the form to get library card :)
```

6

2.Login

```
1  choice: ";
2          cin >> menu;
3          switch (menu)
4          {
5          case 1:
6          {
7              lms.signup();
8          }
9          case 2:
10         {
11             map<int, User> users = loadUsers();
12             pair<int, string> loginDAta = logIn();
13             if (users.count(loginDAta.first) && users[loginDAta.first].password == loginDAta.second)
14             {
15                 isUserLoged = true;
16             }
17             else if (loginDAta.first == 61 && loginDAta.second == "admin")
18             {
19                 lms.adminMenu();
20             }
21             else
22             {
23                 cout << RED << "\nLogin failed. Invalid ID or password.\nlogin again\n"
24                     << RESET;
25             }
26             if (isUserLoged)
27             {
28                 LogedUserMenu(lms);
29             }
30             break;
31         }
```

Code

Output

```
Enter your choice: 2
Fill the form to get library card :)
id: 1187
Password: ****
```

3.All Books

```
1  // show all books
2     void showBooks()
3     {
4          ifstream inFile("books.csv");
5          if (inFile.is_open())
6          {
7              cout << "Book Names:\n";
8              string line;
9              getline(inFile, line);
10             int serial = 1;
11             while (getline(inFile, line))
12             {
13                 stringstream ss(line);
14                 string name;
15                 getline(ss, name, ',');
16                 cout << serial << ". " << name << "\n";
17                 serial++;
18             }
19             inFile.close();
20         }
21         else
22         {
23             cout << "Unable to open the file for reading books.\n";
24         }
25     }
```

Code

Output

```
Enter your choice: 3
Book Names:
1. Paradoxical Sajid
2. Paradoxical Sajid 2
3. Sotry of beginning
4. Data Structure
5. Graphics design
6. Saimum Series
7. Crusade Series
8. Hisnul Muslim
9. Revive Your Heart
```

4. All Categories

```cpp
1  // book categories
2      void showCategory()
3      {
4          ifstream inFile("books.csv");
5          if (inFile.is_open())
6          {
7              set<string> categories;
8              string line;
9              getline(inFile, line);
10             while (getline(inFile, line))
11             {
12                 stringstream ss(line);
13                 string name, isbn, category;
14                 getline(ss, name, ',');
15                 getline(ss, isbn, ',');
16                 getline(ss, category, ',');
17                 categories.insert(category);
18             }
19             inFile.close();
20             int list = 1;
21             cout << "Book Categories:\n";
22             for (const string &category : categories)
23             {
24                 cout << list << ". " << category << "\n";
25                 list++;
26             }
27         }
28         else
29         {
30             cout << "Unable to open the file for reading books.\n";
31         }
32     }
```

Code

Output

```
Enter your choice: 4

all books in our Category:
Book Categories:
1. CSE
2. History
3. Islamic
4. Thriller
```

5.About us

Code

Output

```cpp
// about us
void aboutUs()
{
    cout << GREEN << "\n============================\n          About
    Us\n============================\n\nBurhan Uddin Ashik =>
    223311161\nSohag Mia => 223311162\nAfia Akter =>
    223311163\n\n***************************\n*      LMS Console
    Project\n*      Developed by Varendra University         \n*
    31st Batch E Section Students \n*            Subject: Software
    Engineering\n*      Teacher: Delwar Hossain\n*    Release: November
    2023\n***************************\n"
        << RESET;
}
```

```
Enter your choice: 5

============================
          About Us
============================

Burhan Uddin Ashik => 223311161
Sohag Mia => 223311162
Afia Akter => 223311163

***************************
*     LMS Console Project
*        Developed by Varendra University
*            31st Batch E Section Students
*            Subject: Software Engineering
*      Teacher: Delwar Hossain
*    Release: November 2023
***************************
```

User Logged Menu

```cpp
1
2   // sub main function
3   void logedUserMenu(LMS lms)
4   {
5       cin.ignore(numeric_limits<streamsize>::max(), '\n');
6       int logedMenu;
7       do
8       {
9
10          cout << MAGENTA << "\n\n********************\nWelcome Big & Not Found Library :)\n********************\n\n"
11              << RESET << "1.Show All Books\n2.Show All Category\n3.Show All Publications\n4.Search by isbn or name\n5." << CYAN << "About Us\n"
12              << RESET
13              << "6." << RED << "Exit\n\n"
14              << RESET << "Enter your choice: ";
15          cin >> logedMenu;
16          switch (logedMenu)
17          {
18          case 1:
19          {
20              lms.showBooks();
21              break;
22          }
23          case 2:
24          {
25              lms.showCategory();
26              break;
27          }
28          case 3:
29          {
30              lms.showPublications();
31              break;
32          }
33          case 4:
34          {
35              lms.searchBook();
36              break;
37          }
38          case 5:
39          {
40              lms.aboutUs();
41              break;
42          }
43          case 6:
44          {
45              exitProgram();
46              break;
47          }
48
49          default:
50              cout << RED << "invalid choice! please choose a valid choice :)\n"
51                  << RESET;
52          }
53      } while (logedMenu != 6);
54  }
```

Code

Wrong Info

```
Enter your choice: 2
Fill the form to get library card :)
id: 1176
Password: ****
Login failed. Invalid ID or password.
login again
```

Right Info

```
Enter your choice: 2
Fill the form to get library card :)
id: 1187
Password: ****

********************
Welcome Big & Not Found Library :)
********************

1.Show All Books
2.Show All Category
3.Show All Publications
4.Search by isbn or name
5.About Us
6.Exit
```

Output

1.All Books & 2.Categories

```cpp
1   // show all books
2   void showBooks()
3   {
4       ifstream inFile("books.csv");
5       if (inFile.is_open())
6       {
7           cout << "Book Names:\n";
8           string line;
9           getline(inFile, line);
10          int serial = 1;
11          while (getline(inFile, line))
12          {
13              stringstream ss(line);
14              string name;
15              getline(ss, name, ',');
16              cout << serial << ". " << name << "\n";
17              serial++;
18          }
19          inFile.close();
20      }
21      else
22      {
23          cout << "Unable to open the file for reading books.\n";
24      }
25  }
```

```cpp
1   // book categories
2   void showCategory()
3   {
4       ifstream inFile("books.csv");
5       if (inFile.is_open())
6       {
7           set<string> categories;
8           string line;
9           getline(inFile, line);
10          while (getline(inFile, line))
11          {
12              stringstream ss(line);
13              string name, isbn, category;
14              getline(ss, name, ',');
15              getline(ss, isbn, ',');
16              getline(ss, category, ',');
17              categories.insert(category);
18          }
19          inFile.close();
20          int list = 1;
21          cout << "Book Categories:\n";
22          for (const string &category : categories)
23          {
24              cout << list << ". " << category << "\n";
25              list++;
26          }
27      }
28      else
29      {
30          cout << "Unable to open the file for reading books.\n";
31      }
32  }
```

Code

All Books

```
Enter your choice: 3
Book Names:
1. Paradoxical Sajid
2. Paradoxical Sajid 2
3. Sotry of beginning
4. Data Structure
5. Graphics design
6. Saimum Series
7. Crusade Series
8. Hisnul Muslim
9. Revive Your Heart
```

Categories

```
Enter your choice: 4

all books in our Category:
Book Categories:
1. CSE
2. History
3. Islamic
4. Thriller
```

Output

We Discussed it previous slide

3. All Publications

```cpp
// show all publications
void showPublications()
{
    ifstream inFile("books.csv");
    if (inFile.is_open())
    {
        set<string> publications;
        string line;
        getline(inFile, line);
        while (getline(inFile, line))
        {
            stringstream ss(line);
            string name, isbn, category, publication;
            getline(ss, name, ',');
            getline(ss, isbn, ',');
            getline(ss, category, ',');
            getline(ss, publication, ',');
            publications.insert(publication);
        }
        inFile.close();
        cout << "Book Publications:\n";
        int listCount = 1;
        for (const string &publication : publications)
        {
            cout << listCount << ". " << publication << "\n";
            listCount++;
        }
    }
    else
    {
        cout << "Unable to open the file for reading books.\n";
    }
}
```

Code

Output

```
Enter your choice: 3
Book Publications:
1. Bangla Sahitto Porishod
2. Gurdian
3. Ilm House
4. Samakalin
5. Unknown
```

4. Search by ISBN or Name

```cpp
// search isbn or name
void searchBook()
{
    cout << "\nenter isbn or name: ";
    string item;
    cin.ignore();
    getline(cin, item);
    ifstream inFile("books.csv");
    if (inFile.is_open())
    {
        string line;
        getline(inFile, line);
        bool found = false;
        while (getline(inFile, line))
        {
            stringstream ss(line);
            string name, isbn, category, publication, author, stock;
            getline(ss, name, ',');
            getline(ss, isbn, ',');
            getline(ss, category, ',');
            getline(ss, publication, ',');
            getline(ss, author, ',');
            getline(ss, stock, ',');
            if (name == item || isbn == item)
            {
                cout << GREEN << "Book found: \n"
                    << "Name : " << name << "\n"
                    << "ISBN: " << isbn << "\n"
                    << "Category: " << category << "\n"
                    << "Publication: " << publication << "\n"
                    << "Author: " << author << "\n"
                    << "Stock: " << stock << "\n"
                    << RESET;
                found = true;
                break;
            }
        }
        inFile.close();
        if (!found)
        {
            cout << "No book found.\n";
        }
    }
    else
    {
        cout << "Unable to open the file for reading books.\n";
    }
}
```

Code

Output

Wrong Info

```
Enter your choice: 4

enter isbn or name: 2398523
No book found.
```

Right Info

```
Enter your choice: 4

enter isbn or name: 23489
Book found:
Name : Revive Your Heart
ISBN: 23489
Category: Islamic
Publication: Gurdian
Author: Nouman Ali Khan
Stock: 34
```

5. All Publications

```cpp
// show all publications
    void showPublications()
    {
        ifstream inFile("books.csv");
        if (inFile.is_open())
        {
            set<string> publications;
            string line;
            getline(inFile, line);
            while (getline(inFile, line))
            {
                stringstream ss(line);
                string name, isbn, category, publication;
                getline(ss, name, ',');
                getline(ss, isbn, ',');
                getline(ss, category, ',');
                getline(ss, publication, ',');
                publications.insert(publication);
            }
            inFile.close();
            cout << "Book Publications:\n";
            int listCount = 1;
            for (const string &publication : publications)
            {
                cout << listCount << ". " << publication << "\n";
                listCount++;
            }
        }
        else
        {
            cout << "Unable to open the file for reading books.\n";
        }
    }
```

Code

Output

```
Enter your choice: 3
Book Publications:
1. Bangla Sahitto Porishod
2. Gurdian
3. Ilm House
4. Samakalin
5. Unknown
```

Admin Page

```cpp
// admin menu
int aMenuList()
{
    cout << MAGENTA << "\n\n*********************\n    Admin Panel\n*********************\n\n"
         << RESET << "1.Add Books\n2.Show All Books\n3.Show All Users\n"
         << "4." << YELLOW << "Stock out books\n5."
         << RESET
         << CYAN << "About Us\n"
         << RESET
         << "6." << RED << "Exit\n\n";
    cout << RESET << "Enter your choice: ";
    int adminMenu;
    cin >> adminMenu;
    return adminMenu;
}
void adminMenu()
{
    int adminMenu;
    do
    {
        adminMenu = aMenuList();

        switch (adminMenu)
        {
        case 1:
        {
            addBook();
            break;
        }
        case 2:
        {
            showBooks();
            break;
        }
        case 3:
        {
            printAllUsers();
            break;
        }
        case 4:
        {
            stockOut();
            break;
        }
        case 5:
        {
            aboutUs();
            break;
        }
        case 6:
        {
            exitProgram();
            break;
        }

        default:
            cout << RED << "invalid choice! please choose a valid choice :)\n"
                 << RESET;
        }
    } while (adminMenu != 6);
}
```

Code

Output

```
Enter your choice: 2
Fill the form to get library card :)
id: 61
Password: *****

*********************
      Admin Panel
*********************

1.Add Books
2.Show All Books
3.Show All Users
4.Stock out books
5.About Us
6.Exit
```

1. Add Books

```
1  // add book (admin)
2  void addBook()
3  {
4      cout << "Enter book details:\n";
5      cout << "Name: ";
6      string name;
7      cin.ignore();
8      getline(cin, name);
9
10     cout << "ISBN: ";
11     string isbn;
12     char isbnc;
13     cin >> isbnc;
14     getline(cin, isbn);
```

```
1  // books class
2  class Book
3  {
4  public:
5      string name, isbn, category, publication, author;
6      int stock;
7
8      // Constructor
9      Book(const string &n, const string &i, const string &c,
10         const string &p, const string &a, int s)
11       : name(n), isbn(i), category(c), publication(p), author(a), stock(s) {}
12 };
```

```
52         outFile << newBook.name << "," << newBook.isbn << "," << newBook.category << ","
53                 << newBook.publication << "," << newBook.author << "," << newBook.stock << "\n";
54
55         outFile.close();
56         cout << "Book added and saved to CSV file successfully.\n";
57     }
58     else
59     {
60         cout << "Unable to open the file for saving books.\n";
61     }
62 }
```

Code

Output

```
Enter your choice: 1
Enter book details:
Name: Akasher Opare Akash
ISBN: 238434
Category: Islamic
Publication: Ilm House
Author: Lost Modesty
Stock: 23
Book added and saved to CSV file successfully.
```

2. All Books

```
1  // show all books
2  void showBooks()
3  {
4      ifstream inFile("books.csv");
5      if (inFile.is_open())
6      {
7          cout << "Book Names:\n";
8          string line;
9          getline(inFile, line);
10         int serial = 1;
11         while (getline(inFile, line))
12         {
13             stringstream ss(line);
14             string name;
15             getline(ss, name, ',');
16             cout << serial << ". " << name << "\n";
17             serial++;
18         }
19         inFile.close();
20     }
21     else
22     {
23         cout << "Unable to open the file for reading books.\n";
24     }
25 }
```

We Discussed it previous slide

Code

Output

```
Enter your choice: 2
Book Names:
1. Paradoxical Sajid
2. Paradoxical Sajid 2
3. Sotry of beginning
4. Data Structure
5. Graphics design
6. Saimum Series
7. Crusade Series
8. Hisnul Muslim
9. Revive Your Heart
10. Akasher Opare Akash
```

3.All Users

```cpp
// show all books
void showBooks()
{
    ifstream inFile("books.csv");
    if (inFile.is_open())
    {
        cout << "Book Names:\n";
        string line;
        getline(inFile, line);
        int serial = 1;
        while (getline(inFile, line))
        {
            stringstream ss(line);
            string name;
            getline(ss, name, ',');
            cout << serial << ". " << name << "\n";
            serial++;
        }
        inFile.close();
    }
    else
    {
        cout << "Unable to open the file for reading books.\n";
    }
}
```

Code

Output

```
Enter your choice: 3
List of all users:
1. burhan
2. Ashik
3. Saladin Aiyubi
```

4.Stock out Books

```cpp
// stock out books
void stockOut()
{

    ifstream inFile("books.csv");
    if (inFile.is_open())
    {
        cout << "Books with 0 stock:\n";
        string line;
        getline(inFile, line);
        while (getline(inFile, line))
        {
            stringstream ss(line);
            string name, isbn, category, publication, author;
            int stock;
            getline(ss, name, ',');
            getline(ss, isbn, ',');
            getline(ss, category, ',');
            getline(ss, publication, ',');
            getline(ss, author, ',');
            ss >> stock;
            if (stock == 0)
            {
                cout << YELLOW << "\n"
                     << "Name : " << name << "\n"
                     << "ISBN: " << isbn << "\n"
                     << "Category: " << category << "\n"
                     << "Publication: " << publication << "\n"
                     << "Author: " << author << "\n"
                     << "Stock: " << stock << "\n"
                     << RESET;
            }
        }
        inFile.close();
    }
    else
    {
        cout << "Unable to open the file for reading books.\n";
    }
}
```

Code

Output

```
Enter your choice: 4
Books with 0 stock:

Name : Data Structure
ISBN: 234
Category: CSE
Publication: Unknown
Author: Seymour
Stock: 0

Name : Graphics design
ISBN: 32
Category: CSE
Publication: Unknown
Author: Unknown
Stock: 0
```

```
1  // exit function
2  void exitProgram()
3  {
4       return;
5  }
```

Future Enhancement:

We acknowledge that our current project has many limitations. With these, we will try to improve the project by adding more features in the future. Among the features that I would like to add to this project are:

1. We will bring it to web version in future.

2. Various roles will be added like admin, librarian, general student etc.

3. A student can easily view the issued book and get a reminder to submit the issued book by email.

4. From university student data to authenticate users

 Account creation will be allowed after cross checking the user data.

5. For easy account creation, Student Info scans the QR code on the ID card if implemented.

6. To get the ISBN, the book's QR code scan or bar code scan will have the option to add it easily.

7. Librarians can easily view stock out books.

8. Once the date of return of the issued books is over, the list of those students can be easily viewed.

9. Librarians can view monthly, yearly reports on their dashboard.

10. There will be an admin who can see all library users, books, librarian reports on the dashboard.

Conclusion:

The Library Management System project in C++ is a comprehensive solution for managing library operations. It leverages the power of Object-Oriented Programming (OOP) and efficient data structures provided by the Standard Template Library (STL) in C++. The system is designed to be user-friendly and efficient, ensuring a smooth experience for both librarians and users.

The project stands out due to its comprehensive features, including efficient record keeping, search functionality, automated reminders, and ensuring data consistency and integrity. The use of appropriate software tools like a C++ compiler, an Integrated Development Environment (IDE), a version control system, a UML tool, and a text editor will aid in the successful completion of the project.