**LIBRARY MANAGEMENT SYSTEM**

**OOAD – LAB PROJECT**

**B.E – COMPUTER SCIENCE AND ENGINEERING**

**FROM**

**GOVERNMENT COLLEGE OF ENGINEERING,**

**DHARMAPURI – 636704.**

****­­­

**SUBMITTED BY: -**

**DHINESHKUMAR D (613520104008)**

**SIVASAKTHI V (613520104036)**

**Table of Contents**

1. Problem Statement..................................................................................................

2. Software Requirement Specifications

1. Introduction......................................................................................................................

1.1 Purpose...............................................................................................................

1.2 Scope................................................................................................................

3. Project Description

3.1 Product Perspective

3.1.1 System Interfaces

3.1.2 System Specifications

3.1.2.1 H/W Requirement

3.1.2.2 S/W Requirement

3.1.3 Communication Interfaces

3.1.4 Memory Constraints

3.1.5 Operations

3.1.6 Site Adaption Requirements

3.2 Product functions

3.3 User characteristics

3.4 Constraints

3.5 Assumptions and dependencies

3.6 Apportioning of Requirements

4. Specific Requirements

4.1 External interfaces

4.2 Functions

4.3 Performance requirements

4.4 Logical database requirements

4.5 Design constraints

4.5.1 Standard Compliance

4.6 Software system attributes

4.6.1 Reliability

4.6.2 Availability

4.6.3 Security

4.6.4 Maintainability

4.6.5 Portability

5. Data Dictionary

6. Screens

7. Use Case Diagram

8. Use Case Description

9. Class Diagram

10. DFD Diagram

11. Sequence Diagram

12. Data Design

12.1 Students

12.2 Staffs

12.3 Librarian

12.4 Books

**1.** **PROBLEM STATEMENT**

A college library management is a project that manages and stores books information electronically according to student’s needs. The system helps both students and library manager to keep a constant track of all the books available in the library. It allows both the admin and the student to search for the desired book. It becomes necessary for colleges to keep a continuous check on the books issued and returned and even calculate fine. This task if carried out manually will be tedious and includes chances of mistakes. These errors are avoided by allowing the system to keep track of information such as issue date, last date to return the book and even fine information and thus there is no need to keep manual track of this information which thereby avoids chances of mistakes.

Thus, this system reduces manual work to a great extent allows smooth flow of library activities by removing chances of errors in the details.

**2. SOFTWARE REQUIREMENT SPECIFICATIONS**

**1. INTRODUCTION**

Library Management System is a comprehensive library management solution that is suitable for both large and small libraries. Its flexible design enables Library Management System to be installed in a range of Library organizations, ranging from public libraries, through to academic, joint use and special libraries. This Library Management System Software is capable of handling Books with equal ease and efficiency. This is a Windows-based Library Management System, utilizing the latest advancements in the Information Technology to provide and improve Library Services.

**1.1 Purpose**

The purpose of this project is to develop an application that will automate the whole procedure of a library. The software that would be developed should have facilities like Add / Delete Members, Add / Delete Books, Issue & Return. The application should be secured, as well as with limited access. The main requirement of the project will be the ease of use, besides being the most efficient and effective tool for the purpose. The application should be user friendly. It should be robust and scalable. An automated solution would be very beneficial to the organization, as it would bring structure to the whole process so that it can be traced for any kind of query. Also, an automated solution will lead to optimal utilization of the available resources, reducing duplication of effort, increasing efficiency and minimizing time-delays. Following are the main purpose of computerization:

* To provide services to all the employees for issue, return & search etc. at one place.
* To improve co-ordination in staff.
* To reduce paper filling work.
* To reduce risk of fraud.
* To reduce chances of information leaking.

**1.2 Scope**

For Members: -

* Facility for search of Books based on Author Name or ISBN No.

For Librarian: -

* Simple and intuitive GUI for performing all functions.
* Security features like access control using passwords and username.
* Facility to ADD / DELETE Members, Library Staff & Books and Maintain an easy record of all these.

**3. PROJECT DESCRIPTION**

**3.1 Product Perspective**

The manual library management system includes following drawbacks:

* The existing system involves a lot of paper work and manual calculation. This has led to inconsistency and inaccuracy in the maintenance of data.
* The data, which is stored on the paper only, may be lost, stolen or destroyed due to natural calamity like fire and water.
* The existing system is sluggish and time-consuming causing inconvenience to library staff.
* Since there are many books related to different scopes thus it would be very difficult to find a specific book, or edit the data of some book.

Hence the library management system is proposed with the following Product perspective:

* The computerization of the library system will reduce a lot of paperwork and hence the load on the library staff.
* It would be easy to search a specific book.
* The machine performs all calculations for fines and all. Hence chances of error are nil.
* The system provides for user-ID validation, hence unauthorized access is prevented.

**3.1.1 System Interfaces**

Software will work on Windows OS. The database used will be an open source database like MySQL.

**3.1.2 System Specification**

**Software requirements**

Operating System: we have chosen windows operating system for its best support and user friendliness.

Database: to save the records of the applicants and their details, SQL database is used.

**Hardware requirements**

LMSuses standard Dart classes and databases. So system must need C++ Distributable Frameworks.

**3.1.3 Communication interfaces**

Noadditional specific communication interfaces are needed during the operation of LMS.

**3.1.4 Memory constraints**

The system is expected to have a memory of 256 MB and disk space of 500 MB. But it is recommended that the system has memory capacity of 1 GB and disk space of 1GB.

**3.1.5 Operations**

The basic operations of the ‘Library Management System’ are described as follows:

* The Librarian first has to register him/herself before using the system.
* The Librarian can then login into the system with his/her username and password.
* The Librarian can then add, delete or update the book records within the system.
* The Librarian can issue books when requesting by the members and then update the book record too.
* The Librarian will update the book issue and return records into the book management system.
* The member can search for a specific book by entering book information.
* The Librarian can update member records within the system.

**3.1.6 Site Adaption Requirements**

The system will require an application server for the runtime components and a database for storage. The system will run on select popular application servers and use select popular database for data storage.

**3.2 Product Functions**

There are two different users who will be using this product:

* Librarian who will be acting as the administrator.
* Member who can also use product for search operations.

The features that are available to the Librarian are:

* A librarian can issue a book to the member.
* Can take the book returned from students
* Add books and their information of the books to the database
* Edit the information of the existing books.
* Can access all the accounts of the students.

The features available to the Member are:

* Can search for a particular book

**3.3 User characteristics**

User1: **Librarian**- Librarian will add, delete or update book records within the system. He/she will issue the books as per requested by the member. Staff needs to have complete understanding of functionalities and internal processing of the system.

User2: **Member**- Member will request for book issue and then will return the book. They can search for a specific book. The user does not need to have complete understanding of complex functionalities and internal processing of the system.

**3.4 General Constraints**

Any update regarding the book from the library is to be recorded to have update and correct values and any fine on a member should be notified as soon as possible and should be calculated correctly.

**3.5 Assumptions and Dependencies**

The assumptions are:

* The coding should be error free.
* The system should be user-friendly so that it is easy to use for the users.
* The information of all users, books and libraries must be stored in a database.
* The system should have more storage capacity and provide fast access to the database.

The dependencies are:

* On the basis of listing requirements and specifications the project will be developed.
* The end users should have proper understanding of the product.
* The information of all users must be stored in a database that is accessible by the LMS.
* Any update regarding the book from the library is to be recorded to the database and the data entered should be correct .

**3.6 Apportioning of Requirements**

This is an academic project and hence all the requirements will be completed before the end of the semester. There will be no delay in the delivery of any of the requirements.

**4. SPECIFIC REQUIREMENTS**

**4.1 External Interfaces**

**User Interface**

Various GUI elements like forms, images and standard buttons will be included in the User Interface.

**4.2 Functional Requirements**

* Login:

Description: Librarian will login to the system. The user must be registered in the system before login.

Input: Enter the username and password.

Output: Librarian will be able to use the features of software.

Processing: Username and password will be checked by the system. If they are incorrect a message will be displayed.

* Add/Remove books:

Description: The Librarian can add or remove book by entering details.

Input: Enter the book detail you want to remove or add within the stock.

Output: Confirmation of deletion and update list of books available.

Processing: The details of books must be right in order to add them else there will be problems in future.

* Search:

Description: The users can search a book by entering book details such as author’s name, ISBN No etc.

Input: Enter the details you know about the book.

Output: The list of available books is displayed.

Processing: A message is displayed if the book related to the entered details is not available.

* Issues book:

Description: The Librarian checks the availability of book which the member want to get issued.

Input: Enter Register No of the Member.

Output: Confirmation for book issue or apology for failure in issue.

Processing: If selected book is available then the book will be issued and the record is updated else error will be displayed.

* Return book:

Description: The member wants to return the book.

Input: return the book to the library.

Output: The record will be updated.

Processing: If book is not returned.

**4.3 Performance requirements**

There are no particular extra performance requirements at this point of time.

**4.4 Logical Database Requirements**

Proposed database is intended to store, retrieve, update and manipulate information related to college which include:

* Books availability
* Member details

**4.5 Design Constraints**

Software Constraints: The application shall meet the general standards of Desktop applications.

Hardware Constraints: There is no hardware constraints identified at this point.

**4.5.1 Standard Compliance**

Report Format: All the reports produced for this project are in compliance with the standard templates provided in the class by the advisor.

Naming Conventions: All the documents will be named using the standard naming conventions.

**4.6 Software System Attributes**

**Reliability:**

The application would efficiently store all the information related to the various processes in the system and output the relevant information.

**Availability:**

The application would be available to all the employees of the organizations with an authorized access to the workstations and those who are subject to the authorization permissions.

**Security:**

The system would have adequate security checking through the authentication of the users. The reports would only be available to the employees of the library as per their specific requirements.

**Maintainability:**

The software should not require any additional maintenance. If any errors occur, the user should be able to login again with his credentials. The system shall be flexible enough to add new modules and upgrade the existing modules.

**5. DATA DICTIONARY**

Legal\_character = [a-z|A-Z]

Legal digit = [0-9]

Special\_character = [@|$|#]

Book\_Details = isbnno + book\_name +author\_name +publishedyear+ publisher + edition +

copies

ISBN No = (legal digit) \*

Book\_name = (legal character) \*

Author\_name = (legal character) \*

Publisher = (legal character) \*

Published Year = (Year Format)

Edition = (legal character) \*

Copies = (legal digit) \*

Student\_details = name+member\_id+gender+dob+email+department+aadharno+

username+password

Name=first\_name + last name

first\_name = (legal character) \*

member\_id= (legal digits) \*

Staff\_details = name+member\_id+gender+dob+email+department+aadharno+

username+password

staff\_id = (legal digit) \*

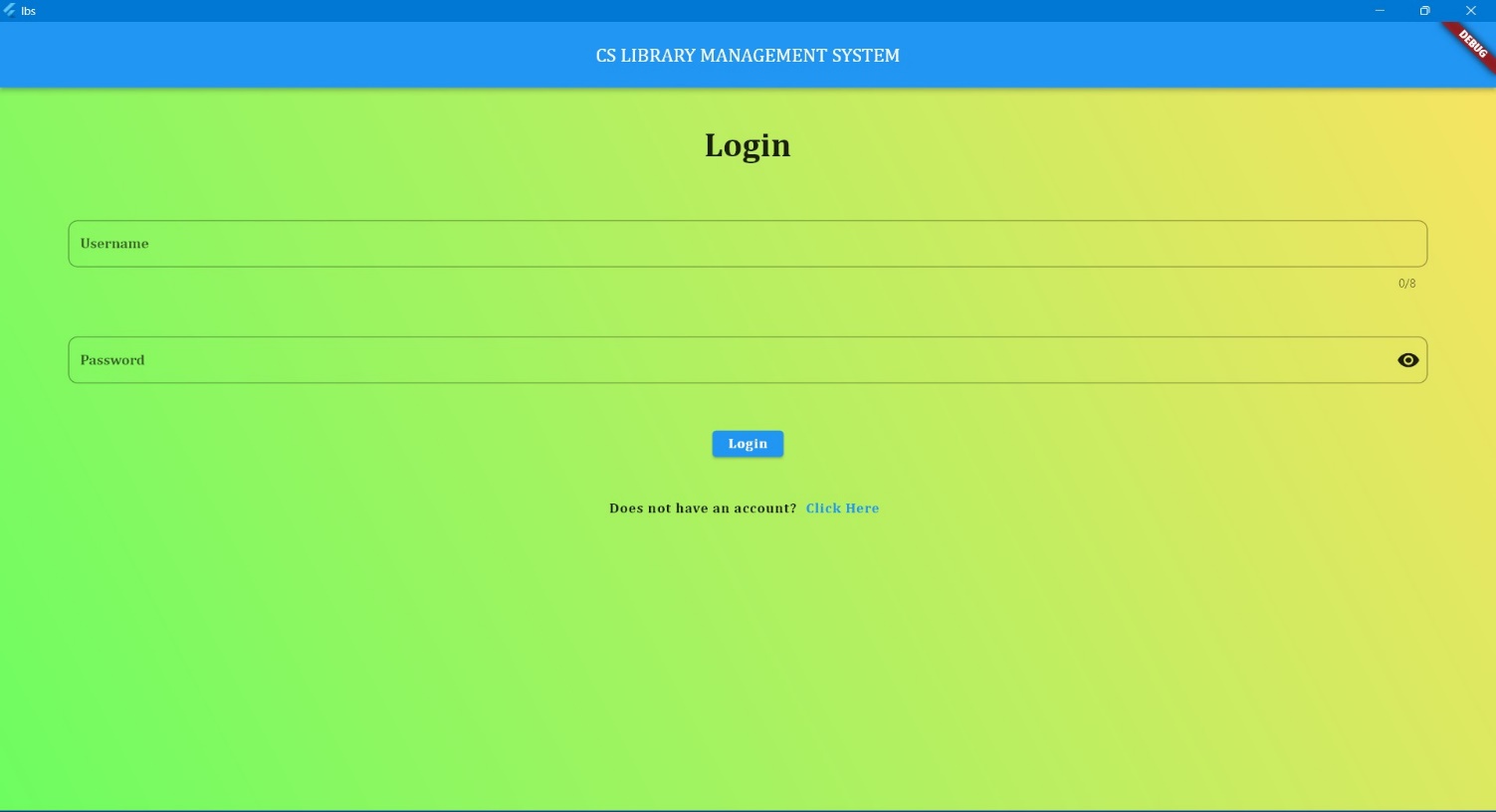
login=username +password

username = (legal character + legal digit) \*

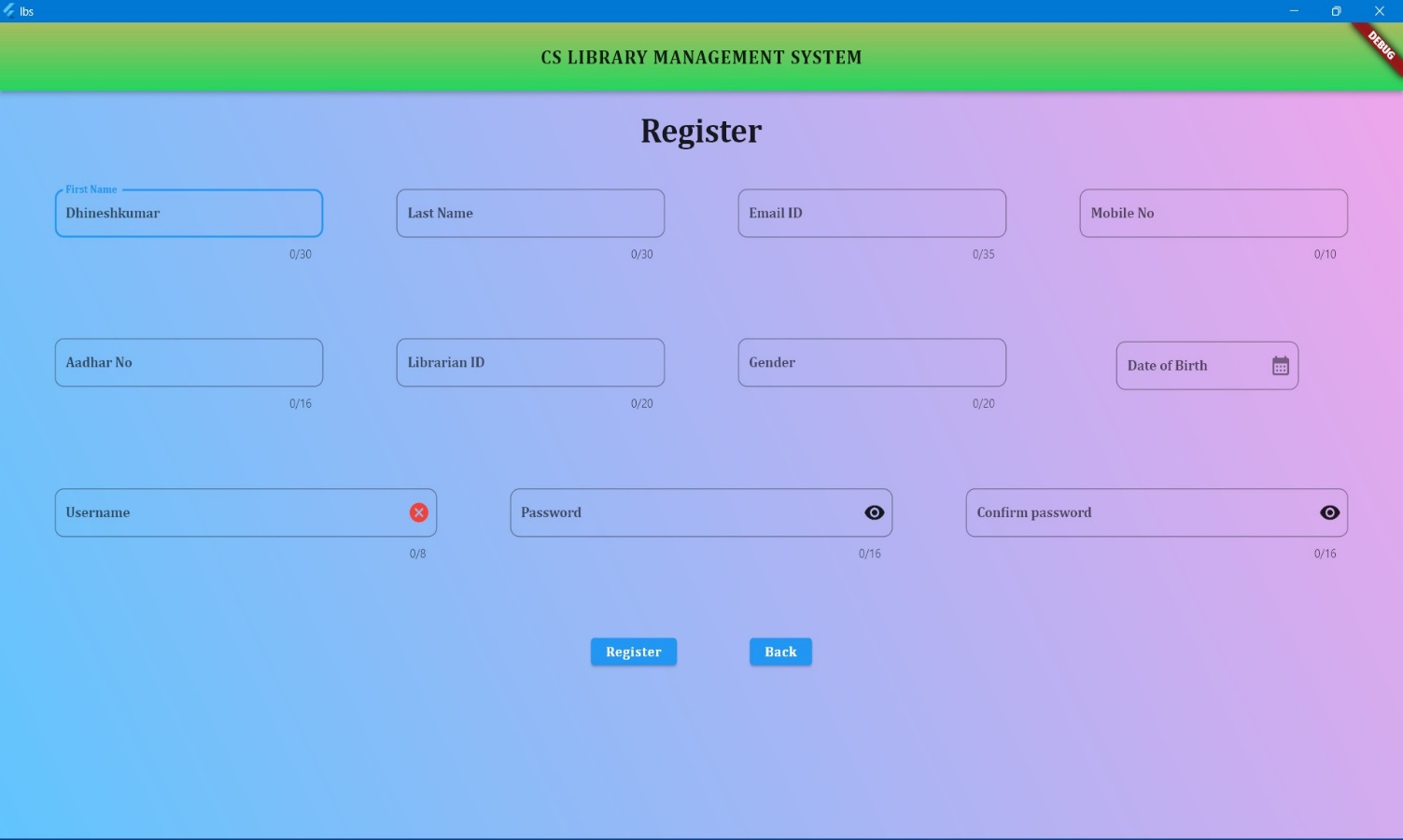
password = (legal character + legal digit) \*

**6. SCREENS:**

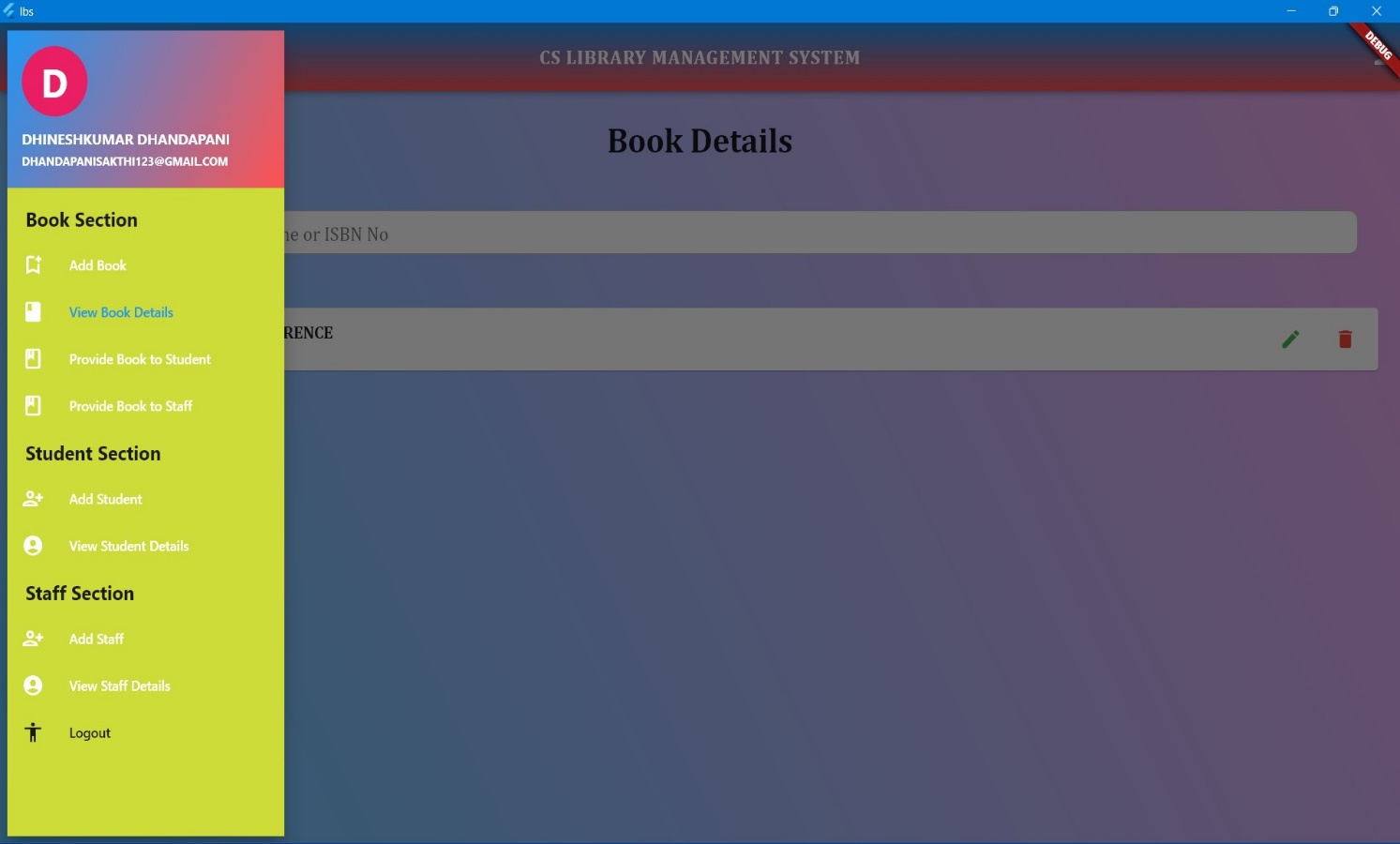
* 1. **LOGIN**

****

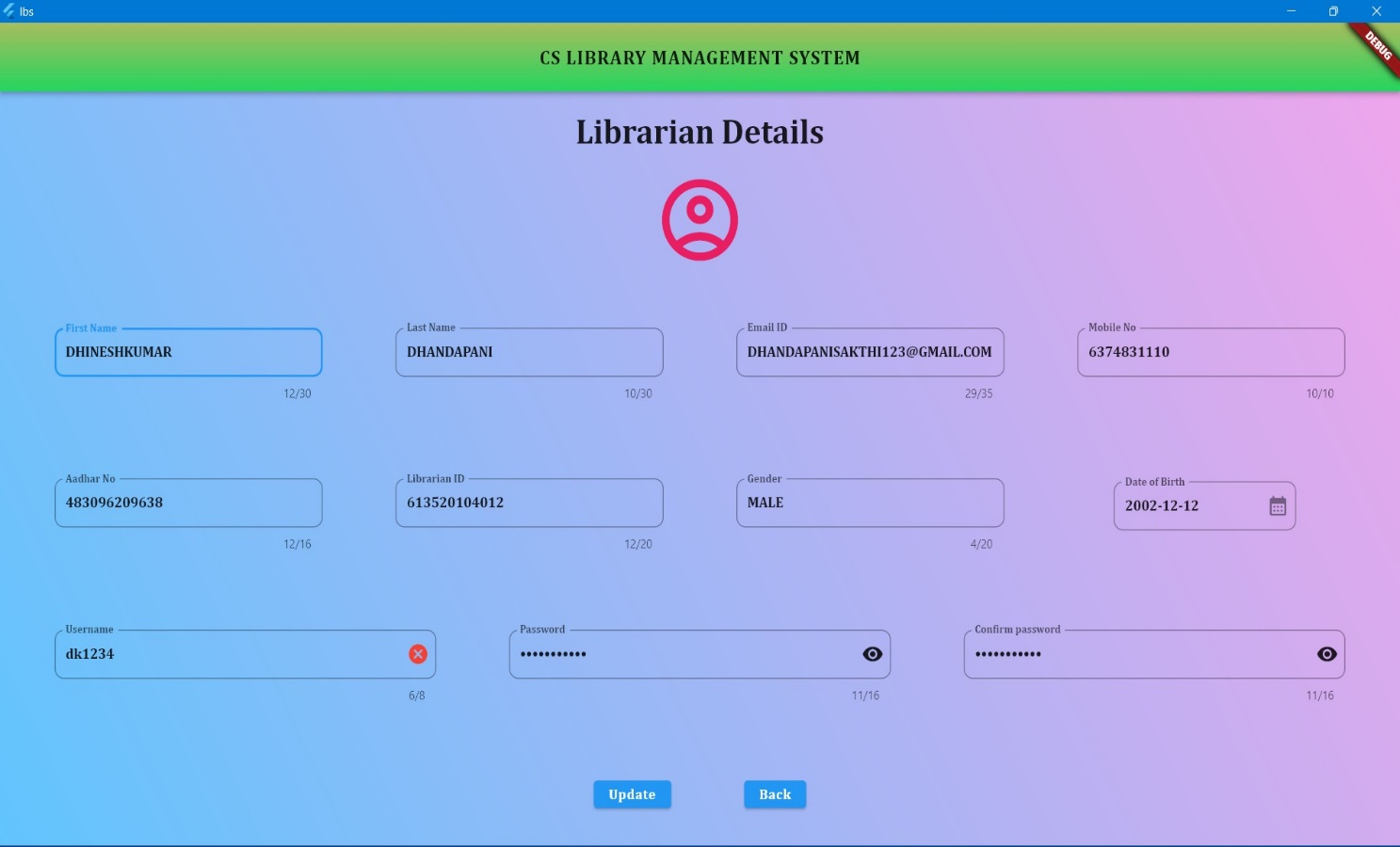
* 1. **REGISTER LIBRARIAN**

****

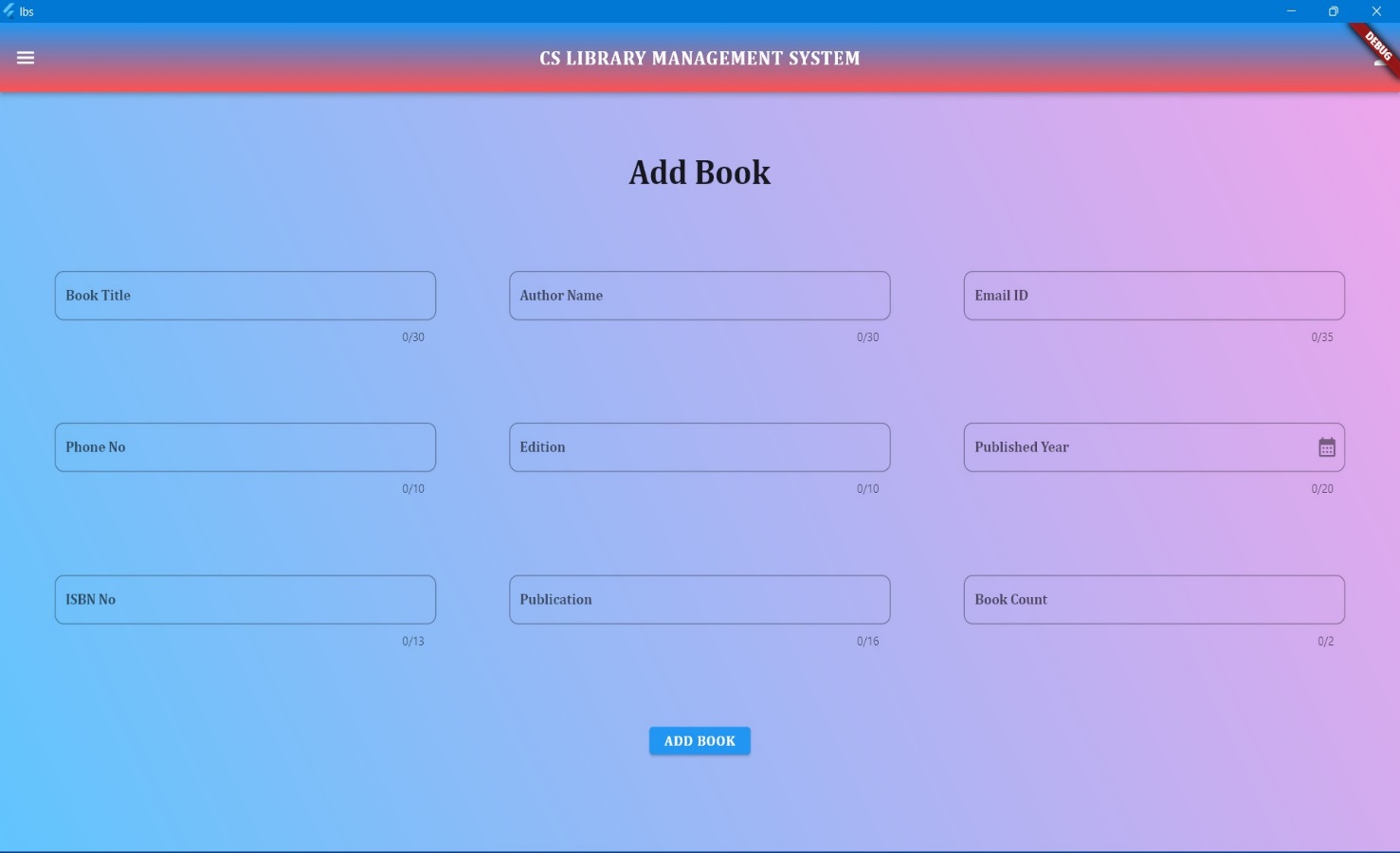
* 1. **HOMEPAGE**

****

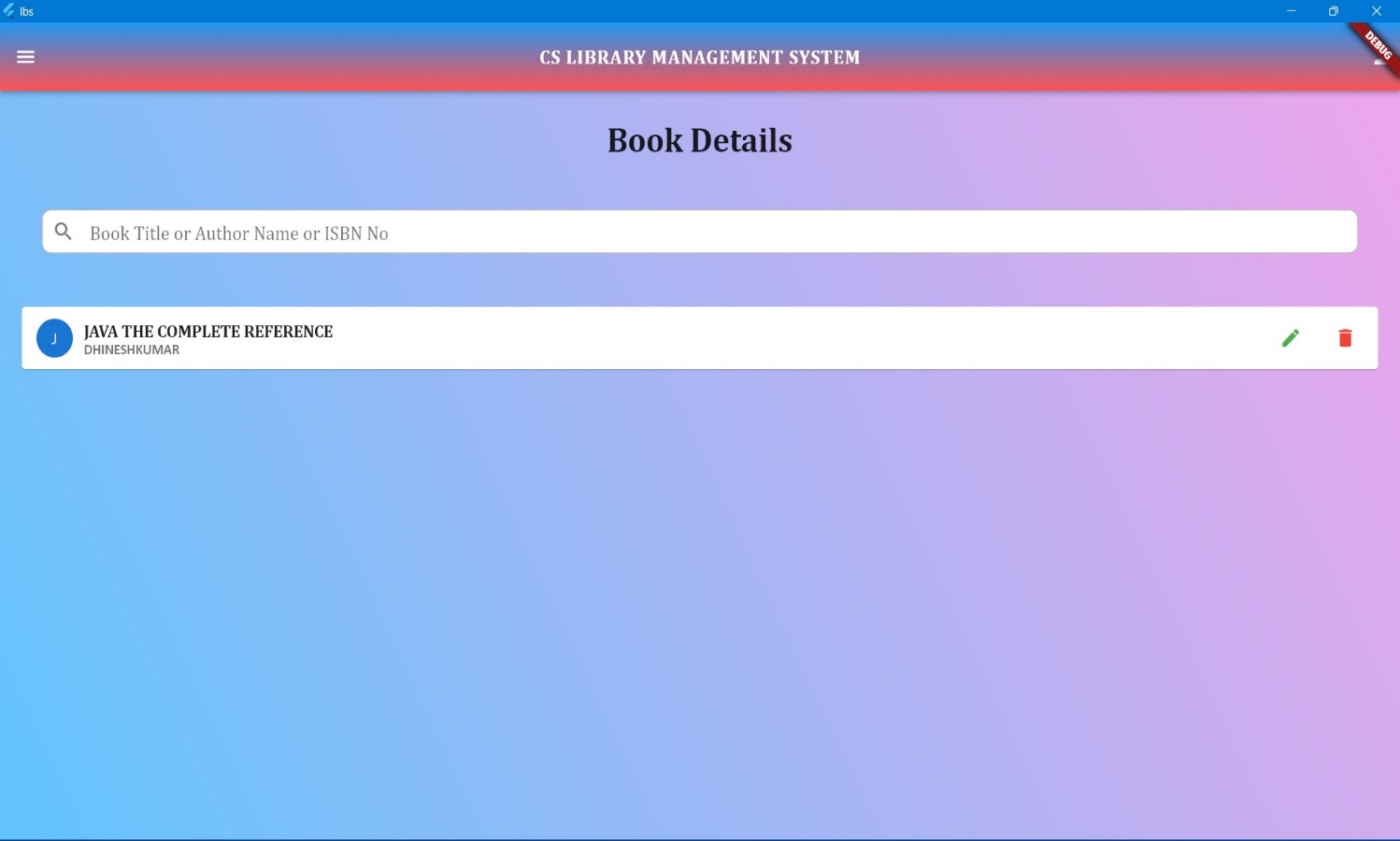
* 1. **UPDATE & VIEW LIBRARIAN DETAILS**

****

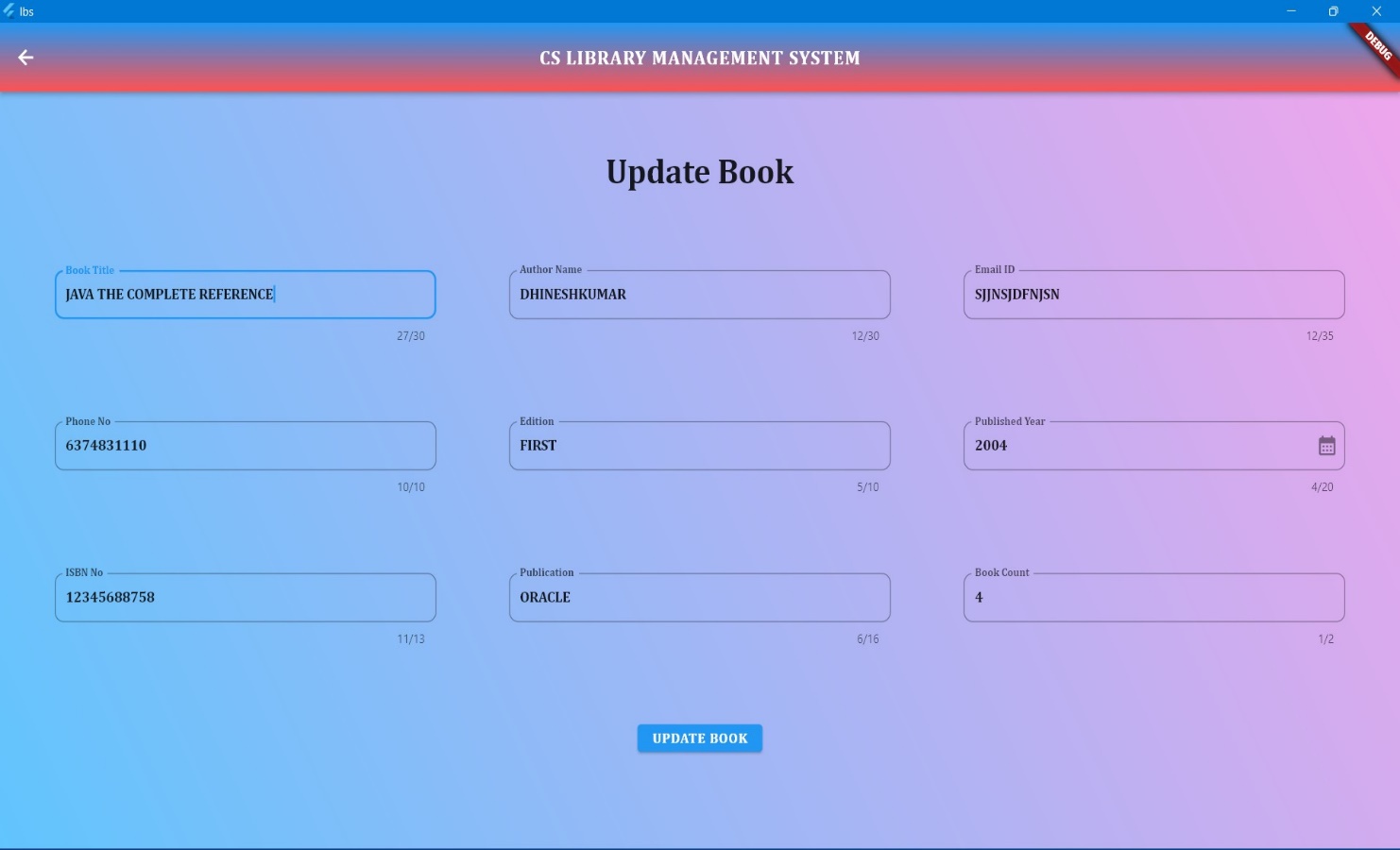
* 1. **ADD BOOK DETAILS**

****

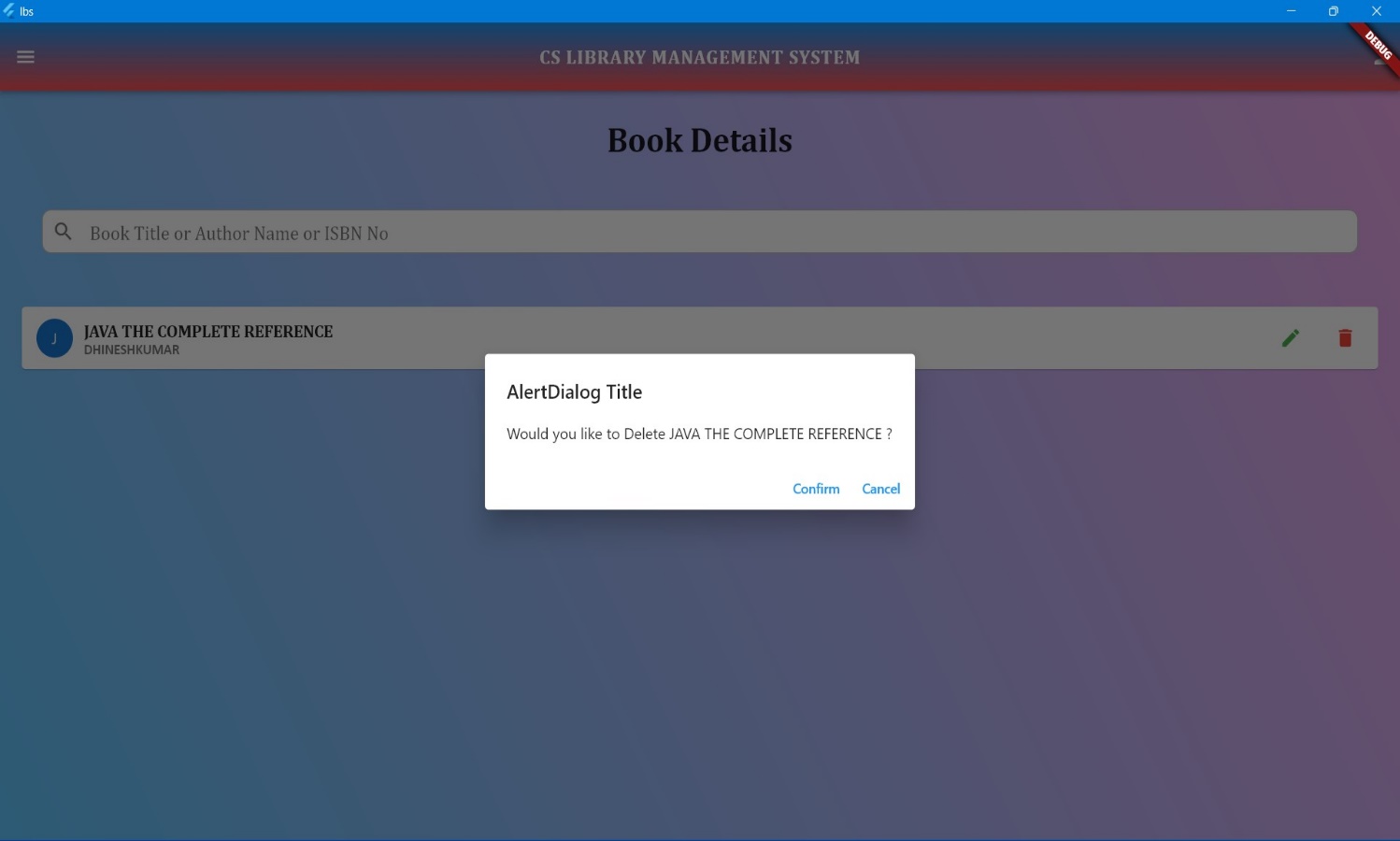
* 1. **VIEW BOOK DETAILS**

****

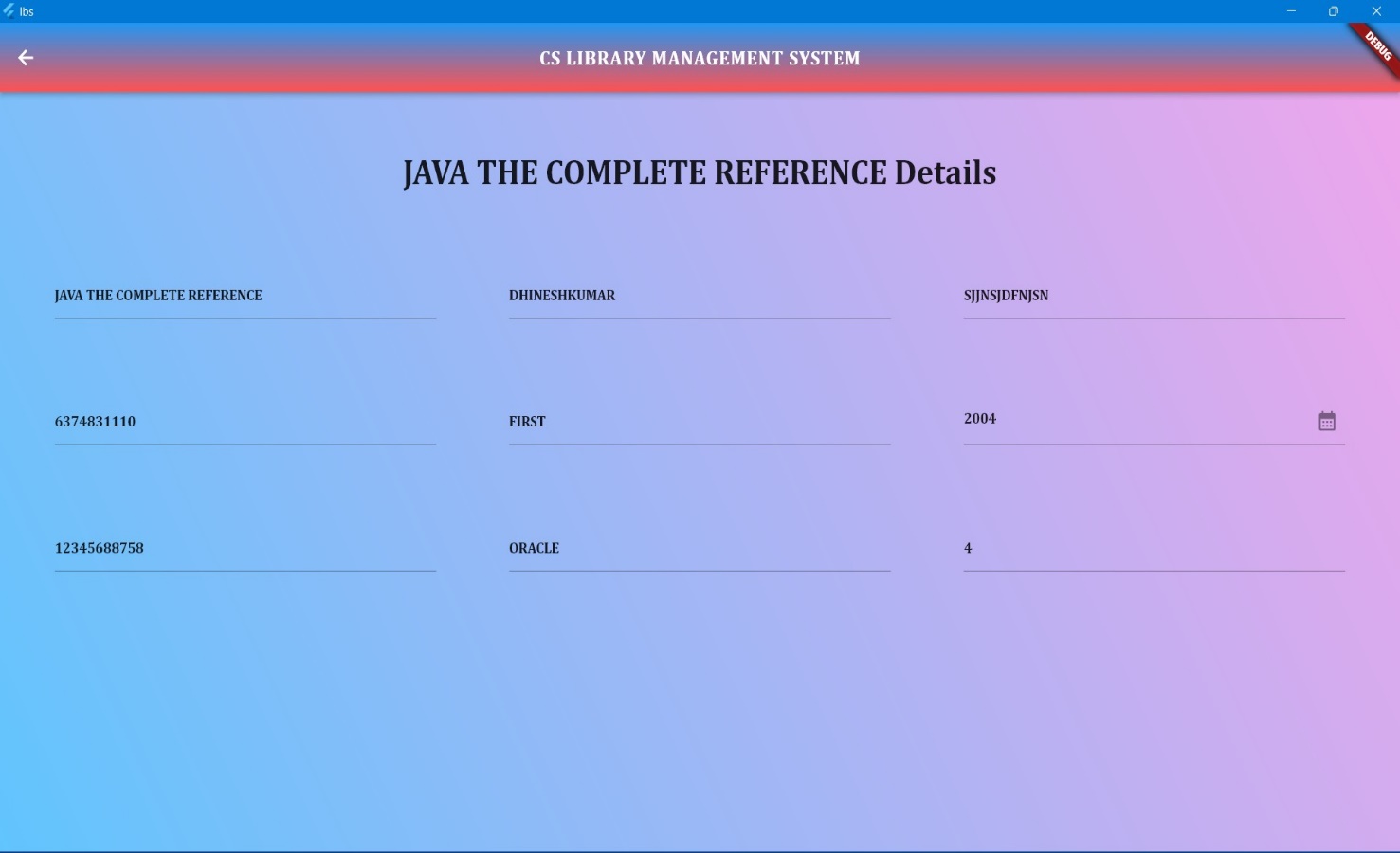
* 1. **UPDATE BOOK DETAILS**

****

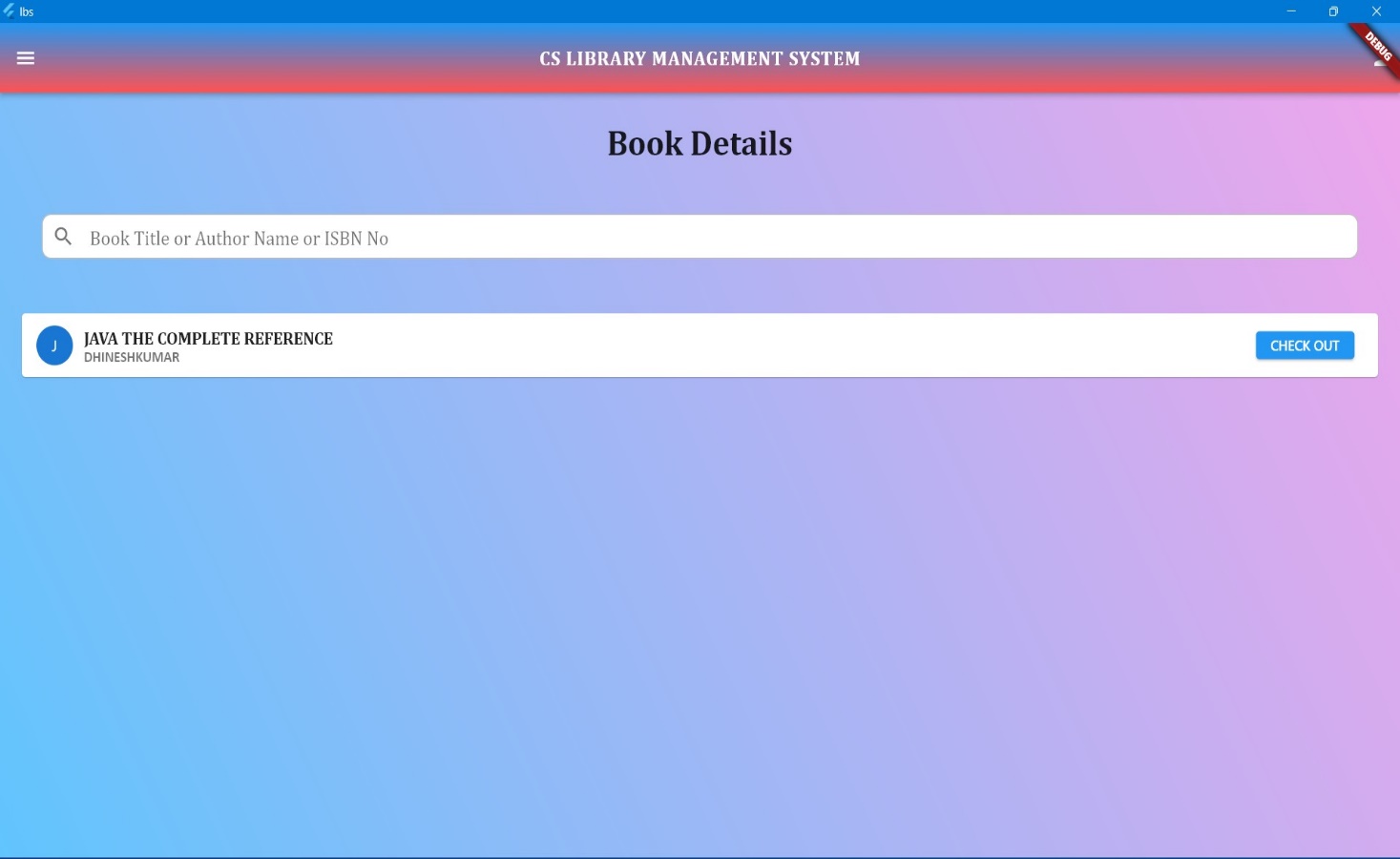
* 1. **DELETE BOOK DETAILS**

****

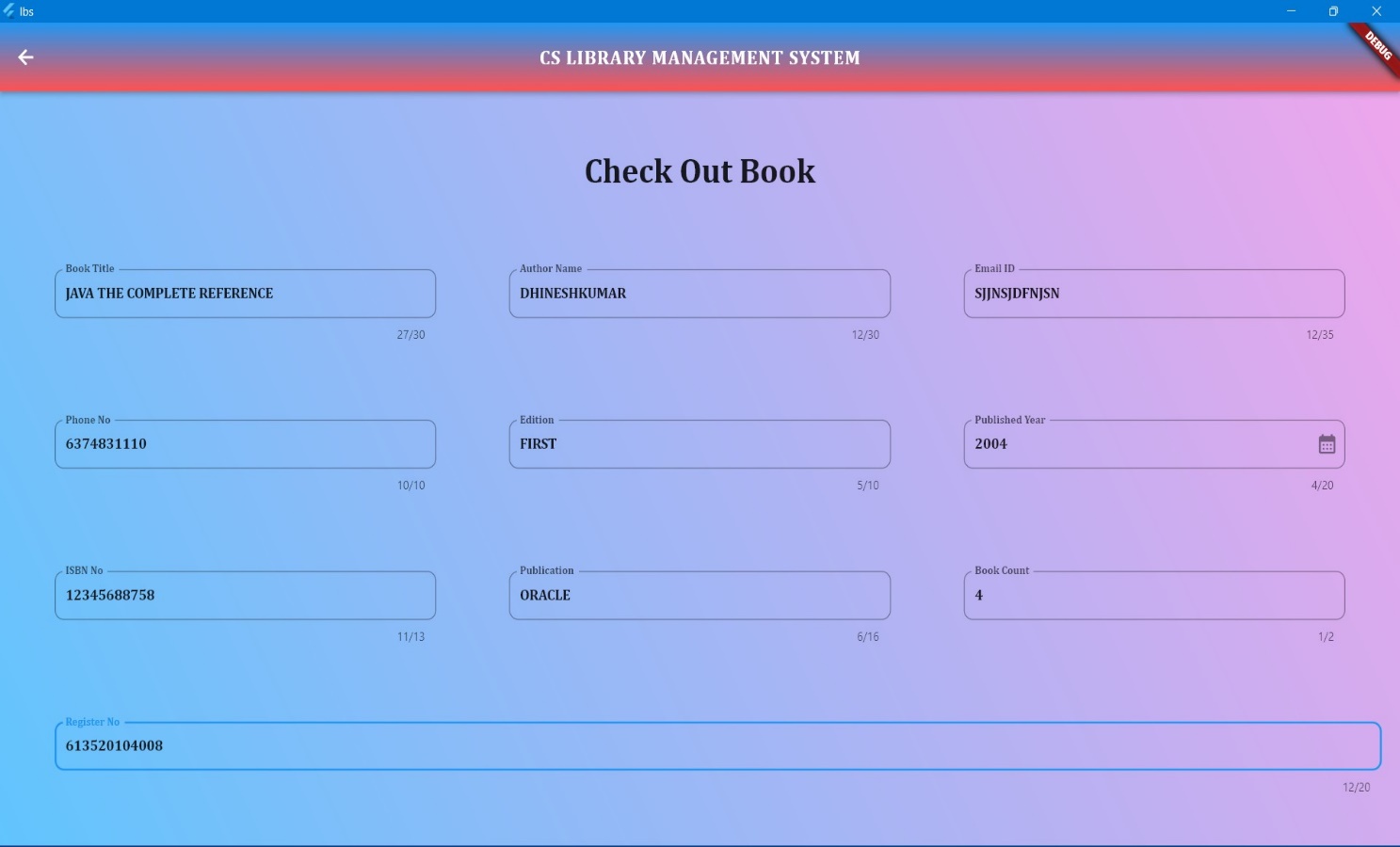
* 1. **VIEW SPECIFIC BOOK DETAILS**

****

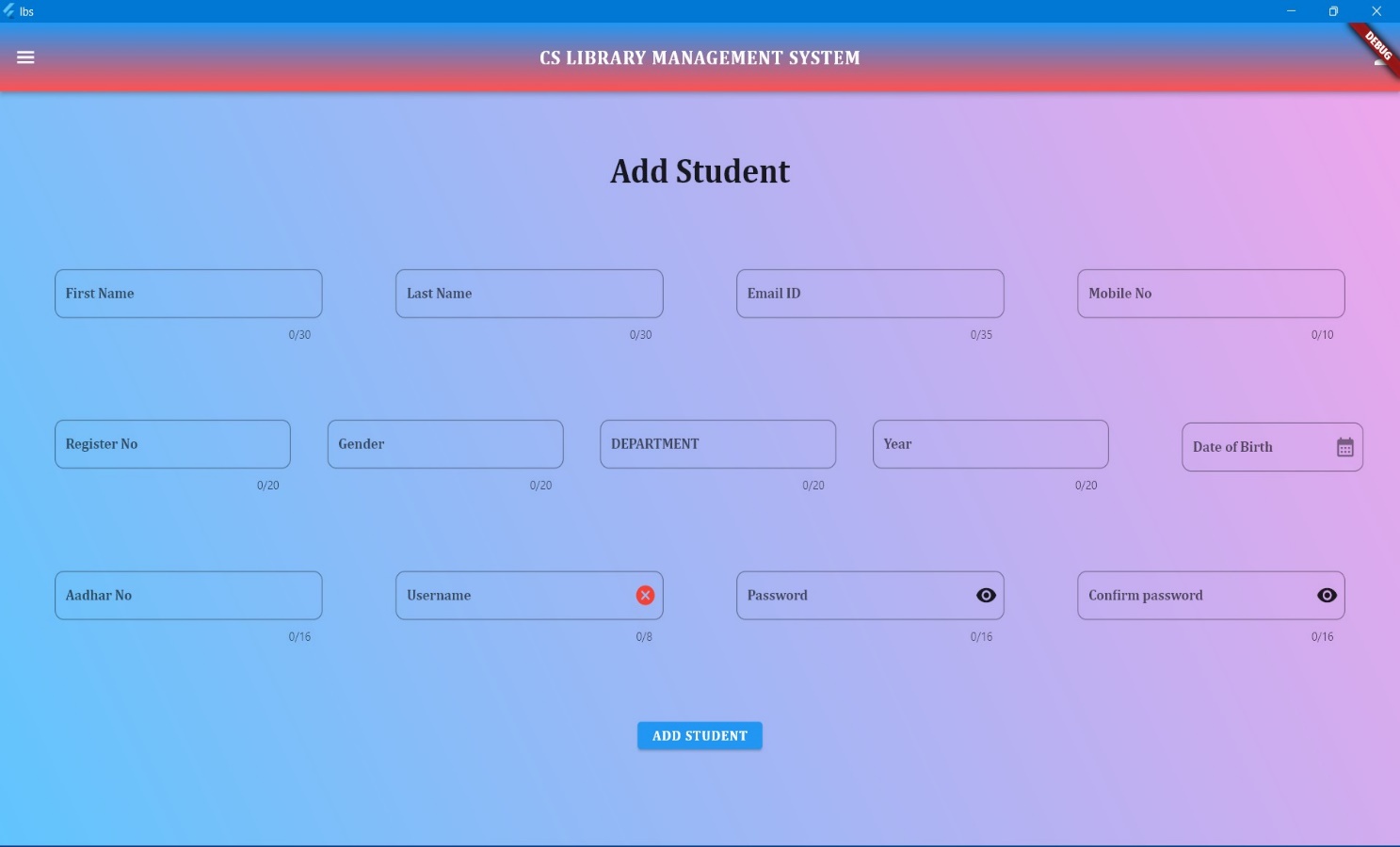
* 1. **PROVIDE BOOK TO STUDENT AND STAFF**

****

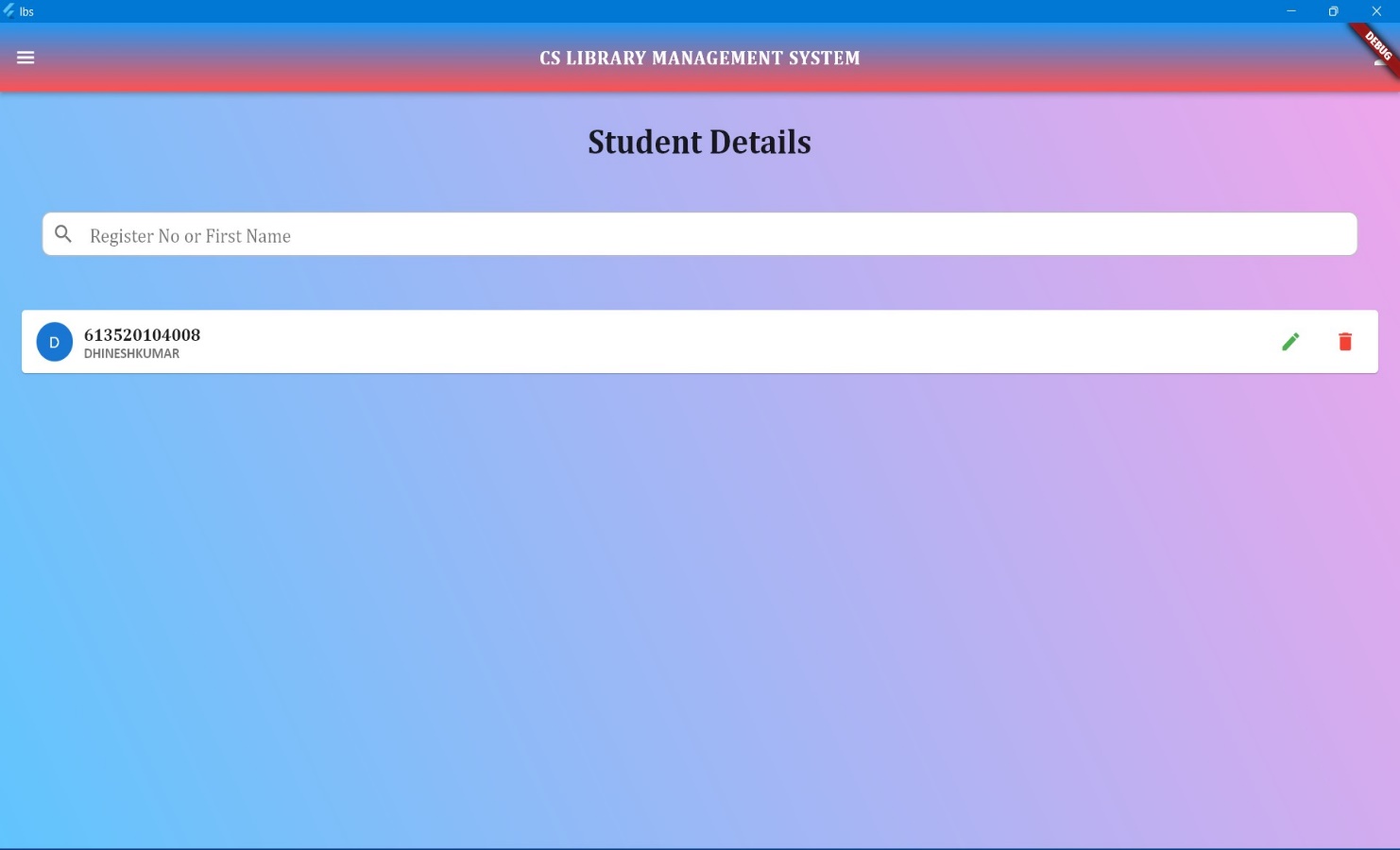
* 1. **CHECK OUT BOOK TO STUDENT**

****

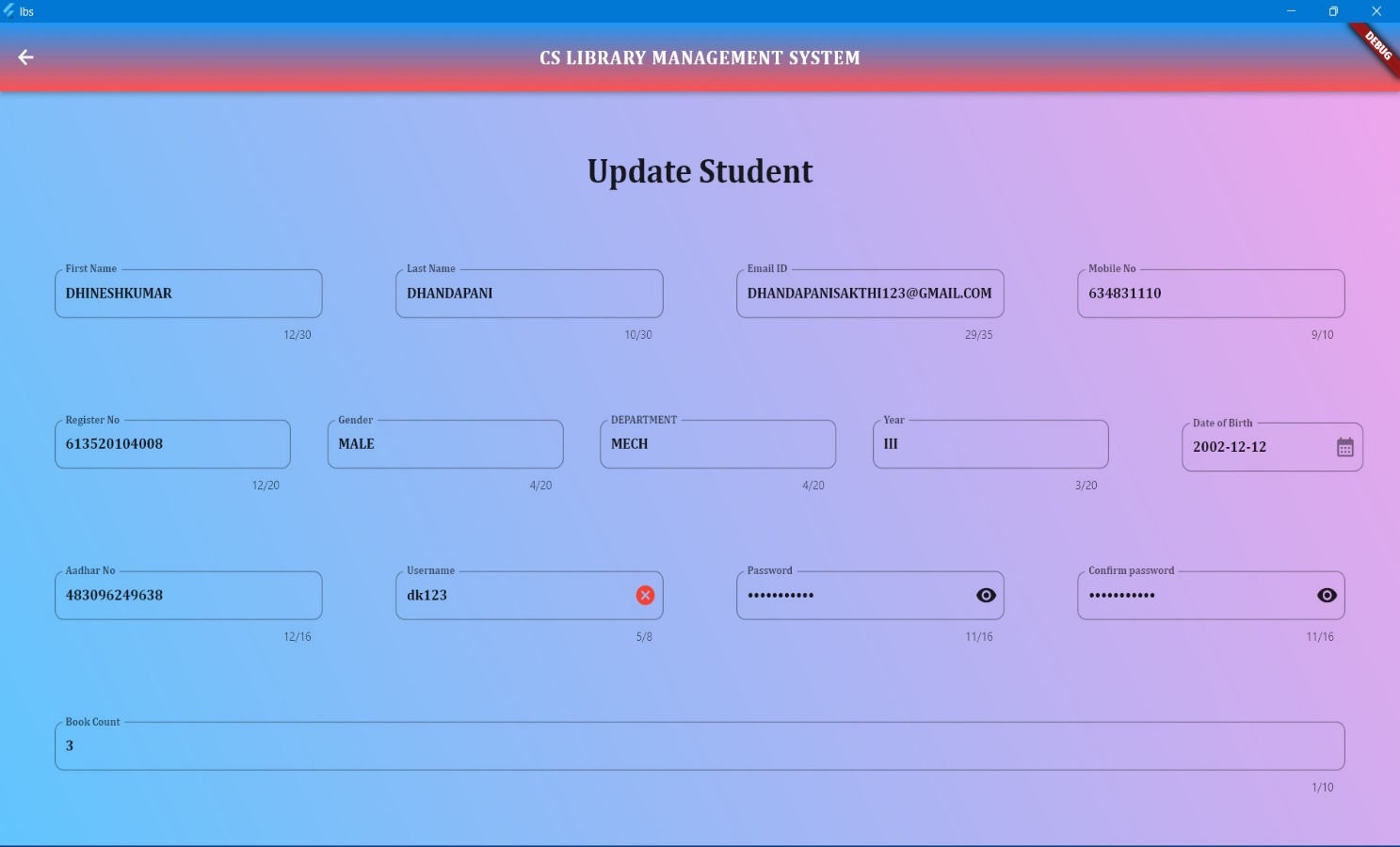
* 1. **ADD STUDENT DETAILS**

****

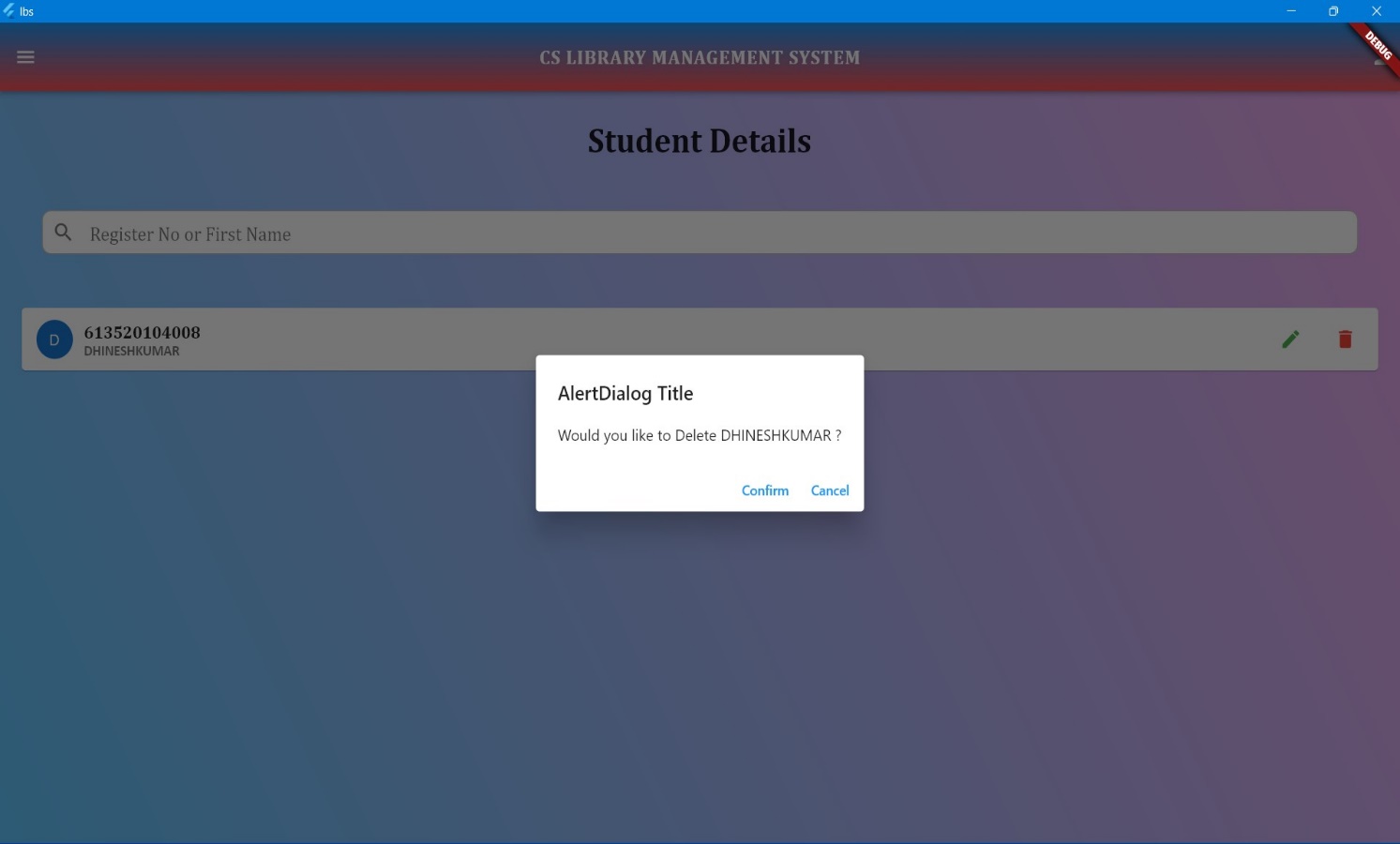
* 1. **VIEW STUDENT DETAILS**

****

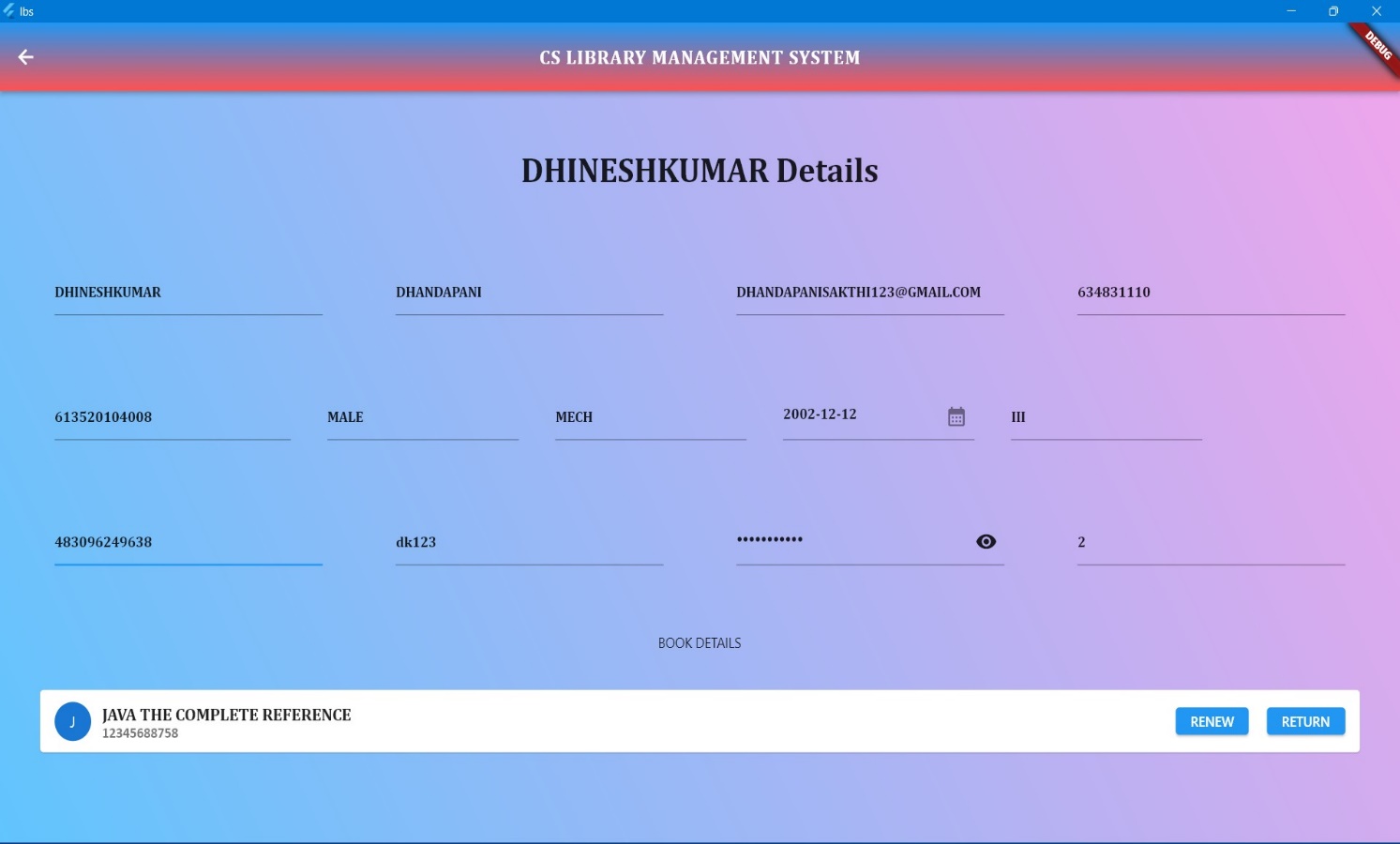
* 1. **UPDATE STUDENT DETAILS**

****

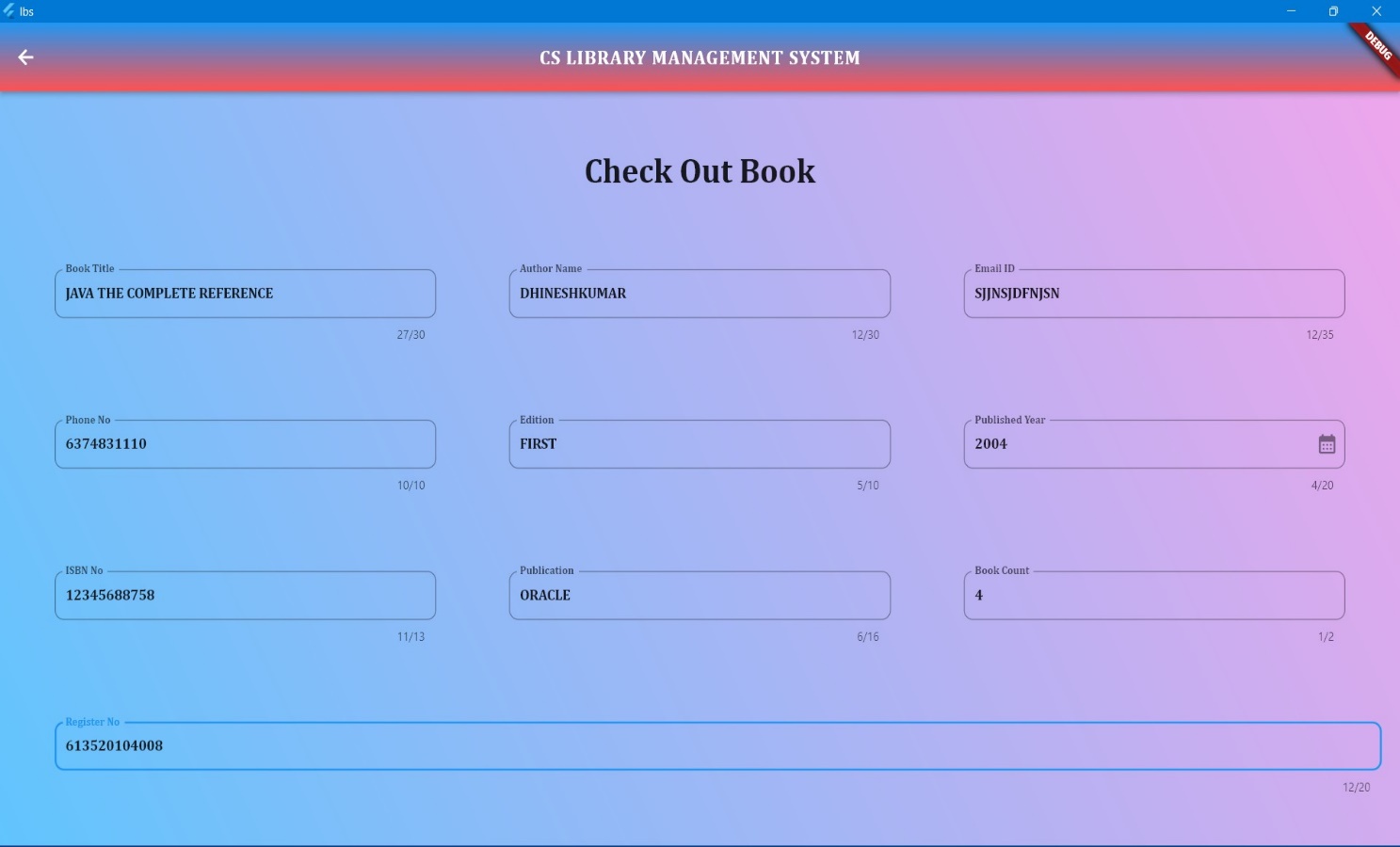
* 1. **DELETE STUDENT DETAILS**

****

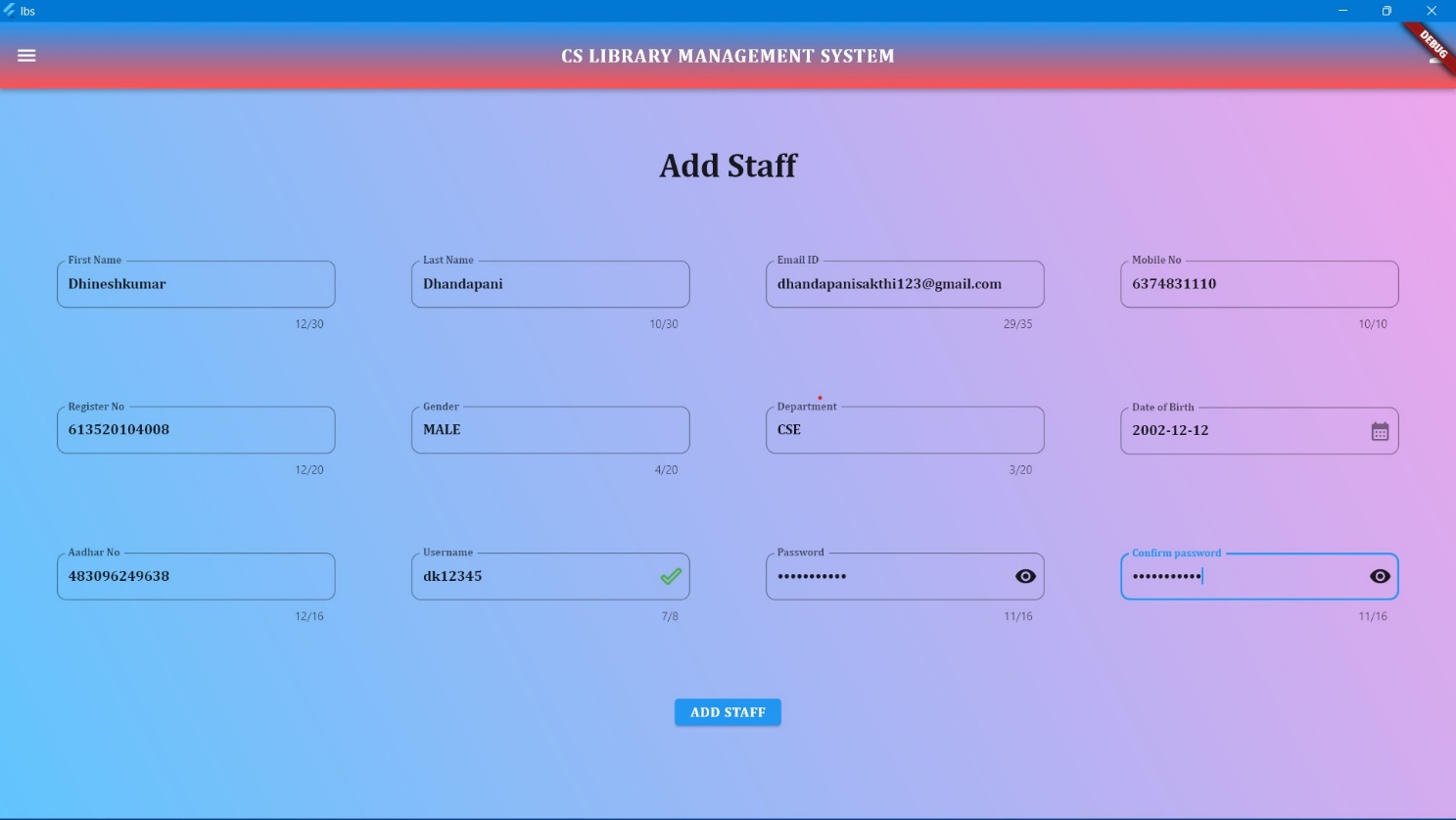
* 1. **VIEW PARTICULAR STUDENT DETAILS**

****

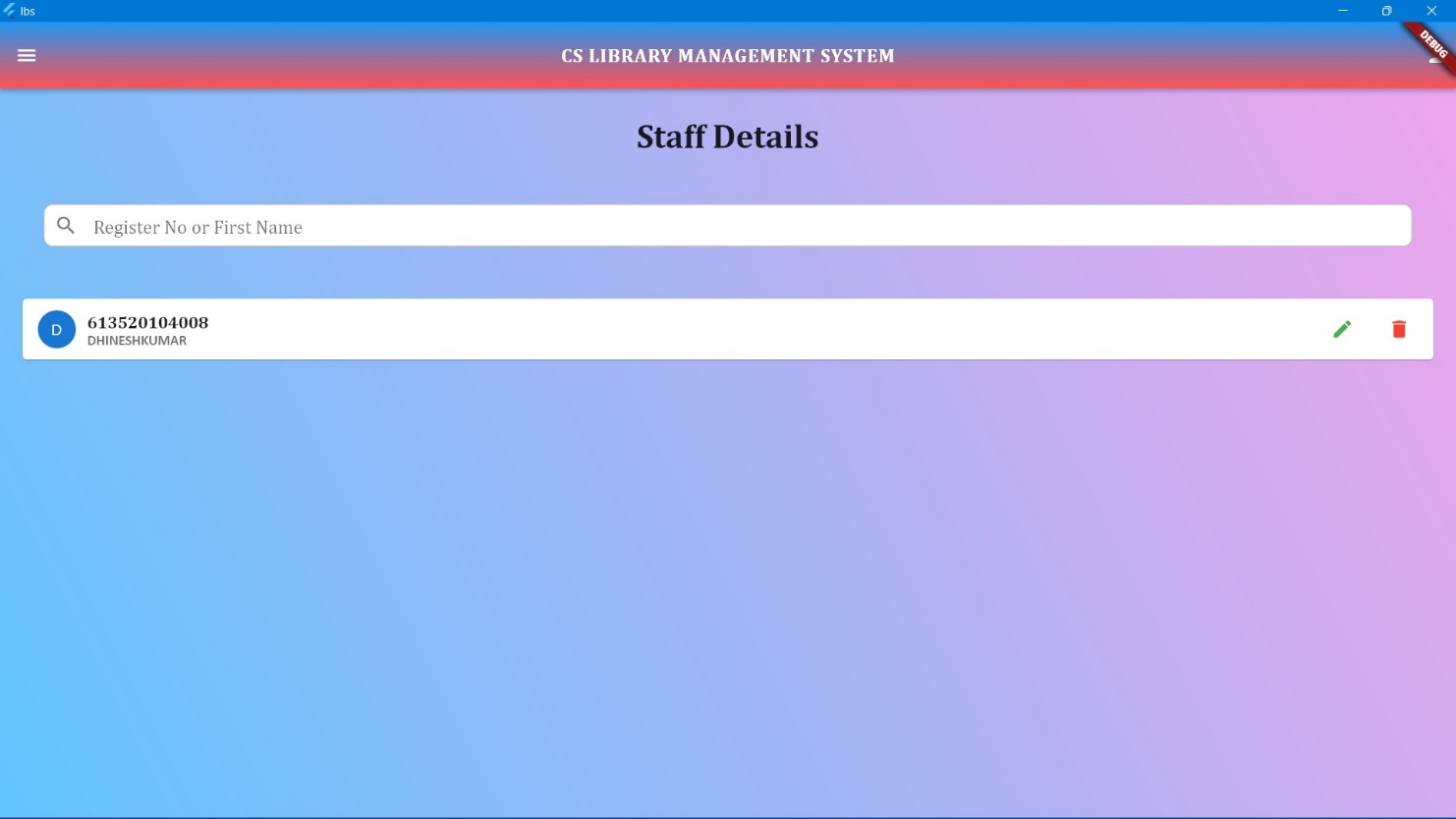
* 1. **CHECK OUT BOOK TO STUDENT**

****

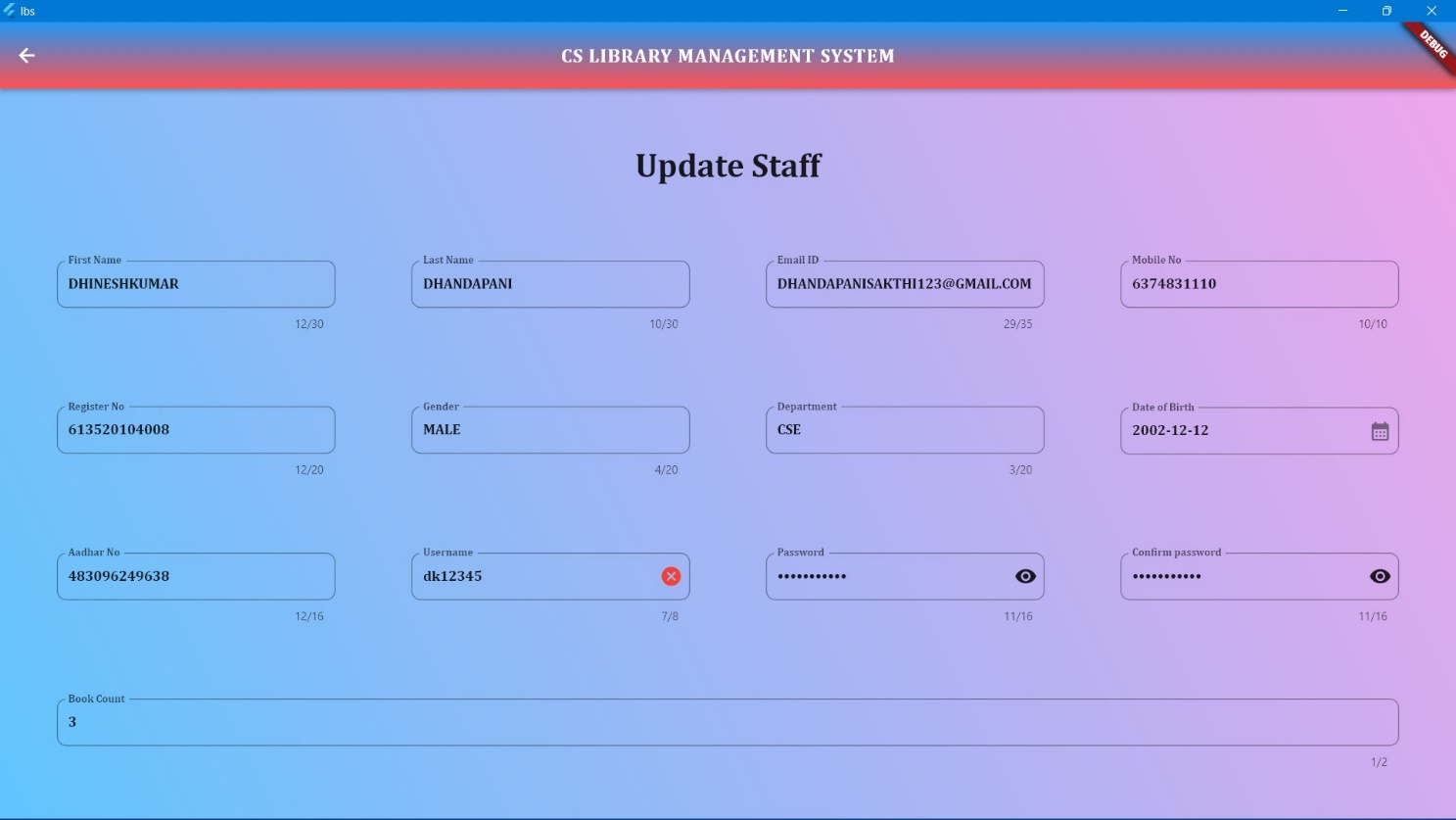
* 1. **ADD STAFF DETAILS**

****

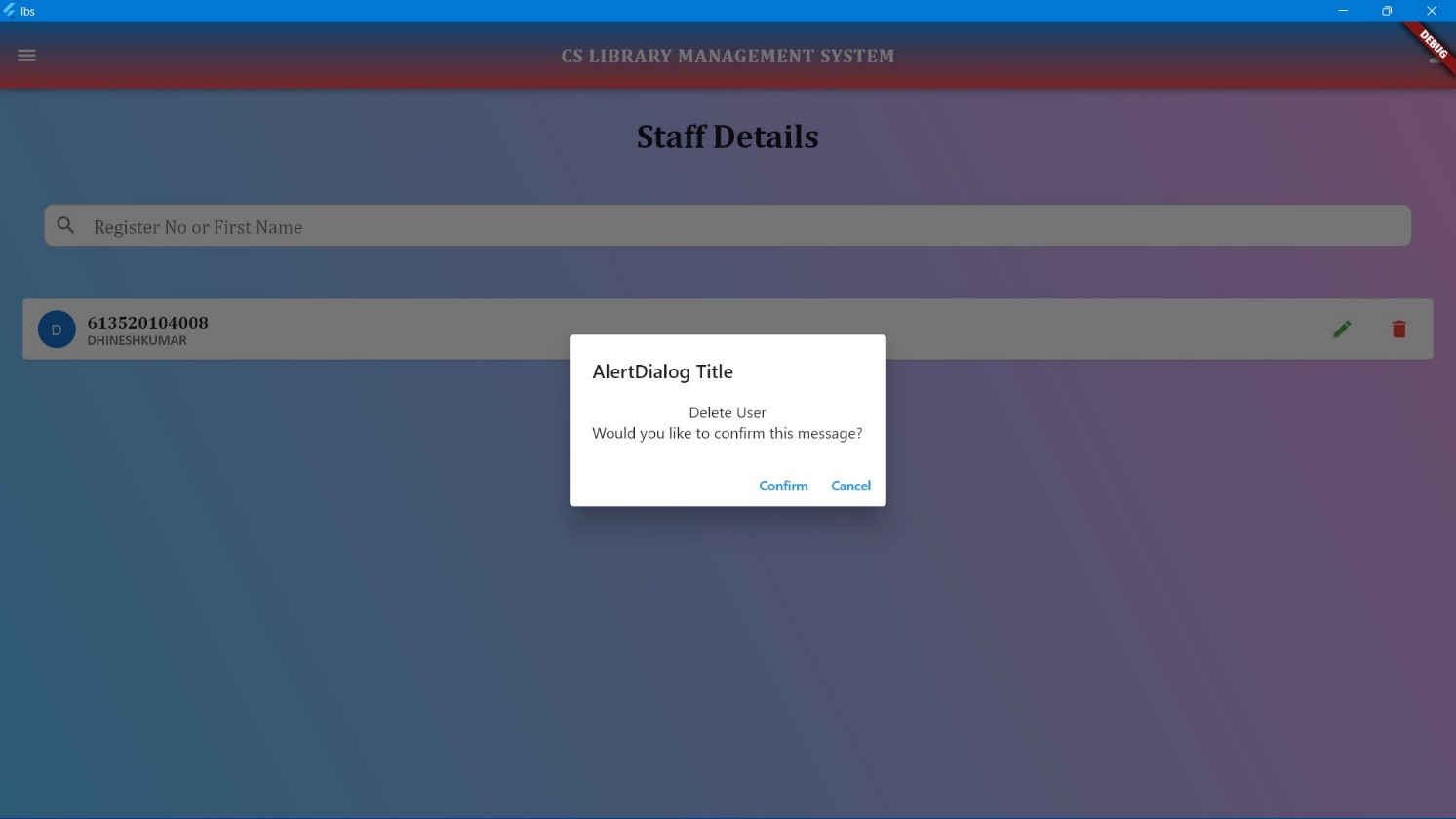
* 1. **VIEW STAFF DETAILS**

****

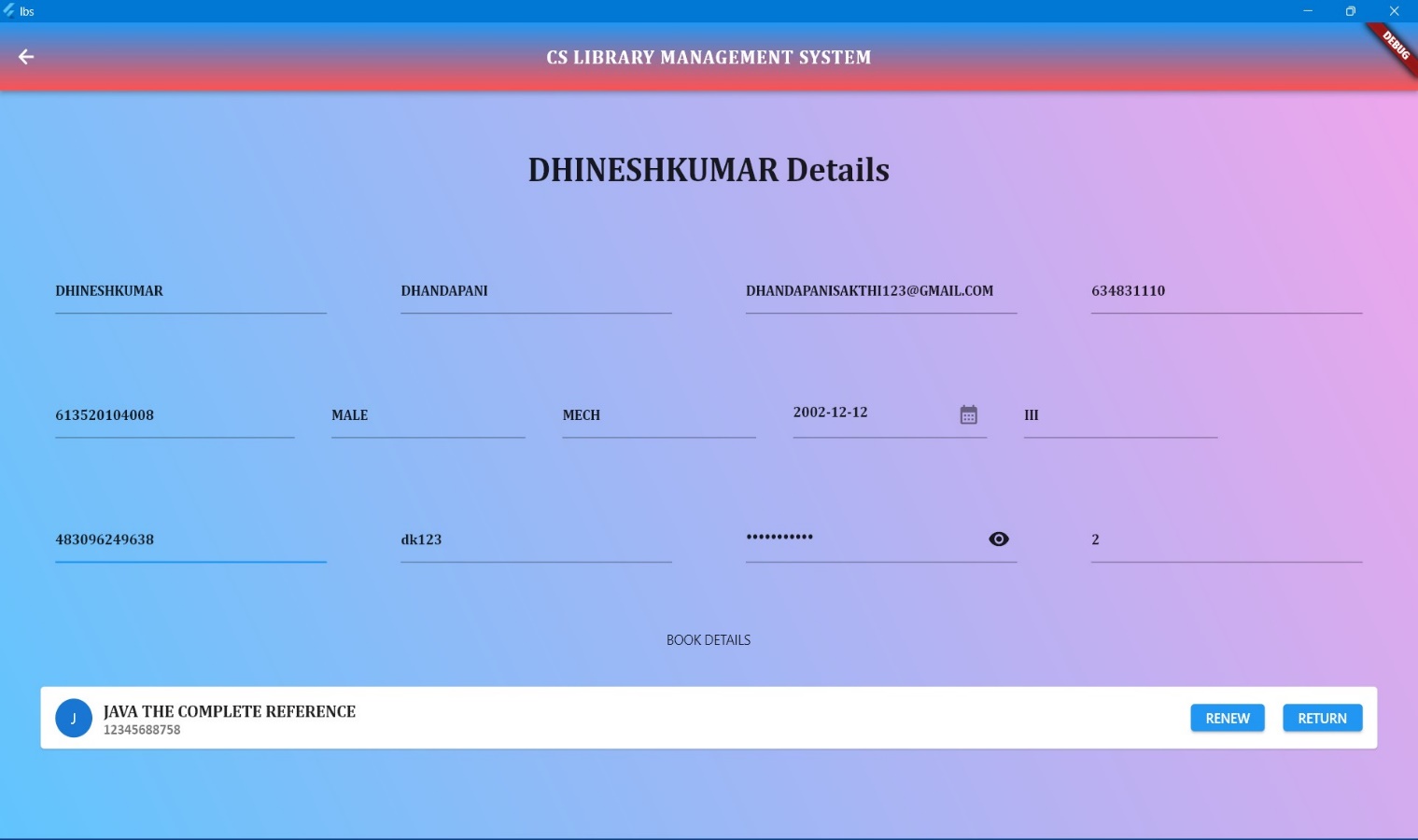
* 1. **UPDATE STAFF DETAILS**

****

* 1. **DELETE STAFF DETAILS**

****

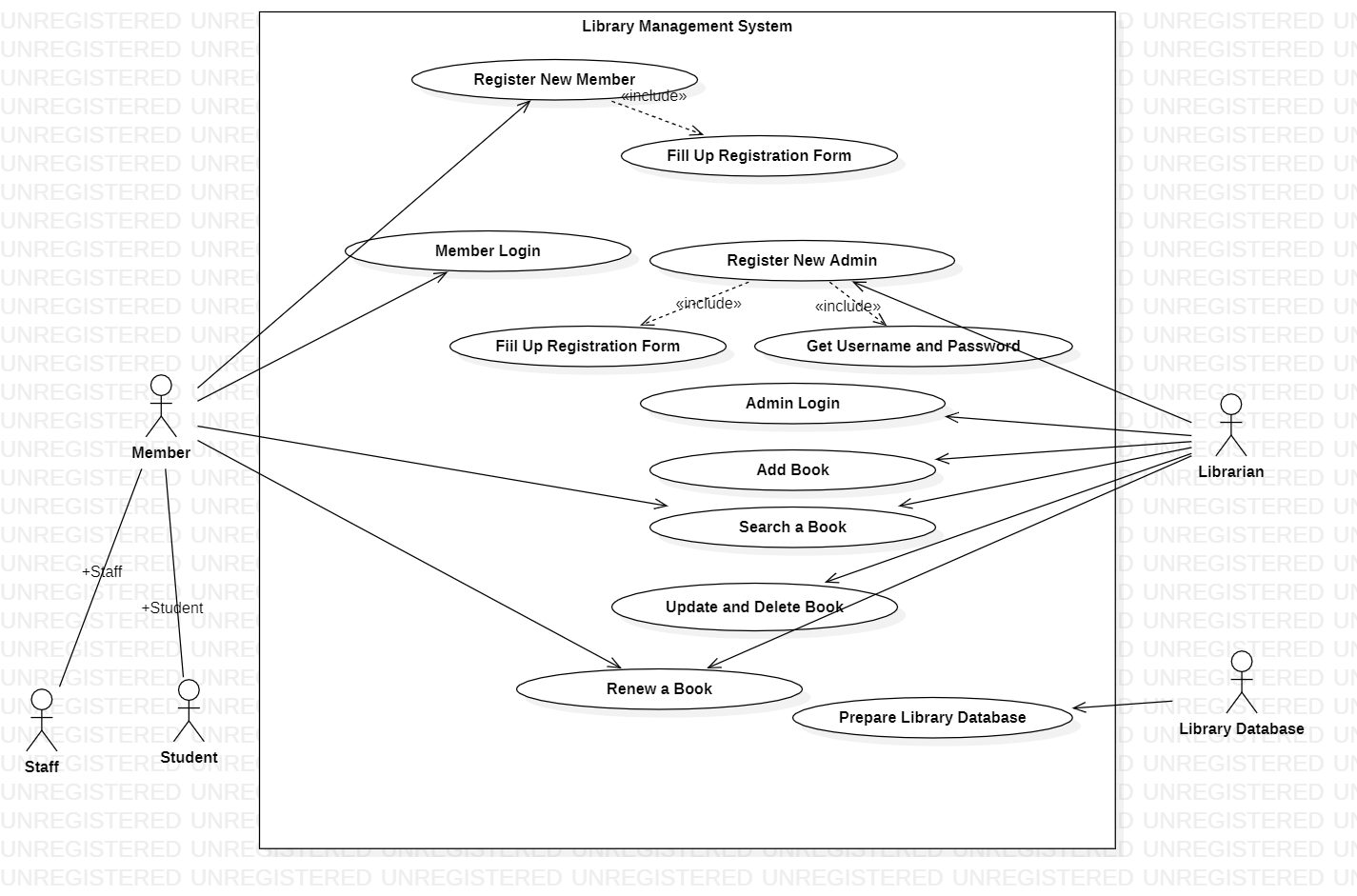
* 1. **VIEW PARTICULAR STAFF DETAILS**

****

**NOTE :**

* RENEW and RETURN BOOKS Facilities are available in the view students and staff Section.

**7. USE CASE DIAGRAM**

****

**8. USECASE DESCRIPTION**

**LOGIN**

**1. Brief Description:**

This use case describes how an actor login into the ‘Library Management System’.

**2. Actors:**

Librarian

**3. Flow of events:**

**a.** Basic Flow:

This use case starts when the actor wishes to log into the ‘Library Management System’.

The system requests the actor to enter the username and password.

The actor enters his/her username and password.

The system validates the username and password and then the actor is logged into the system.

**b.** Alternative flows:

If in the basic flow, the actor enters an invalid name or password, the system displays an error message. The actor can use to either return to the beginning of the basic flow or cancel the login at the point where use case ends.

**4. Special requirements:**

None

**5. Pre-conditions:**

The actor must have an account created in the system prior to executing the use cases.

**6. Post conditions:**

If the use case was successful, the actor is logged in to the system. If not, the system state is unchanged.

**7. Extension points:**

None.

**ADD/DELETE MEMBER**

**1. Brief Description:**

This use case describes how an actor adds or deletes the user’s record in the system.

**2. Actors:**

Librarian

**3. Flow of events:**

**a.** Basic Flow:

This use case starts when the actor has to add or delete users/members within the ‘Library Management System’.

The actor fills the details of the member.

The member is then added to the system now the member got registered for further uses.

The actor can delete a member by filling the appropriate details.

**b.** Alternative flows:

If in the basic flow the actor fills wrong details while deleting a member then the system displays an error message. The actor can either return to the beginning of the basic flow or can cancel the deletion of the member from the system.

**4. Special requirements:**

None.

**5. Pre-conditions:**

The member must added into the system before the actor deletes him/her.

**6. Post conditions:**

Record for a member has been added.

**7. Extension points:** None.

**ISSUE BOOK**

**1. Brief Description:**

This use case describes how the staff issues book when requested by the member.

**2. Actors:**

Librarian

Members

**3. Flow of events:**

**a.** Basic Flow:

If a member wants to borrow a book it is important that the staff should login to the system.

If login is successful the staff should enter the member id to be searched.

If the member search is successful the staff should enter the book id.

If the book is available then it can be borrowed.

**b.** Alternative flows:

If the login fails then the staff should re- register themselves.

If the member search is unsuccessful then the staff should re-register the student.

If the book search is unsuccessful and book data is not found then the staff must enter the book in requisition report.

**4. Special requirements:**

None.

**5. Pre-conditions:**

To borrow any books it is important that the member is registered and the book to be borrowed is available.

**6. Post conditions:**

Book is reserved for the member.

**7. Extension points:**

None.

**MAINTAIN MEMBER RECORD**

**1. Brief Description:**

This use case describes how the actor maintains the record of members which includes edit or view the member’s data.

**2. Actors:**

Librarian

**3. Flow of events:**

**a.** Basic Flow:

Librarian login to the system and selects the menu option to change the data of specific member.

Enter the name of categories that he/she want to change.

The system save the change in membership record and update previous record.

To view the record of a member the actor selects from the menu option.

LMS presents the record of members.

**b.** Alternative Flows

If the password is incorrect then a message is printed on the screen and staff member is returned to the beginning.

If the name of changed to be category is not among the existing categories a message is printed on the screen and the actor is returned to the menu screen.

**4. Special requirements:**

None.

**5. Pre-conditions:**

Password or username must have registered.

**6. Post conditions:**

Member record is updated in database.

**7. Extension points:**

None.

**SEARCH A BOOK**

**1. Brief Description:**

This use case describes how the actor can search for a particular book.

**2. Actors:**

Member

Librarian

**3. Flow of events:**

**a.** Basic Flow:

Member or Librarian enters the or ISBN or author name and presses search

If the search is successful then that book is displayed on the screen.

**4. Special requirements:**

None.

**5. Pre-conditions:**

The book to be searched should be registered into the database of the ‘Library Management System’.

**6. Post conditions:**

List of books according to search data is appeared on the screen.

**7. Extension points:**

None.

**ADD/DELETE BOOK**

**1. Brief Description:**

This use case describes how the actor adds or removes the books.

**2. Actors:**

Librarian

**3. Flow of events:**

**a.** Basic flow:

The Librarian  login to the system.

If login is successful then to add a book the staff must search for the book.

If the book is not currently available and is part of the requisition list it can be added to the book database.

To remove a book it is again searched in the library system.

If it is found it should be checked in borrower record.

If it is not in the borrowed list it can be removed.

**4. Special requirements:**

None.

**5. Pre-conditions:**

To add any book that should be part of requisition list and to delete the book must be part of library.

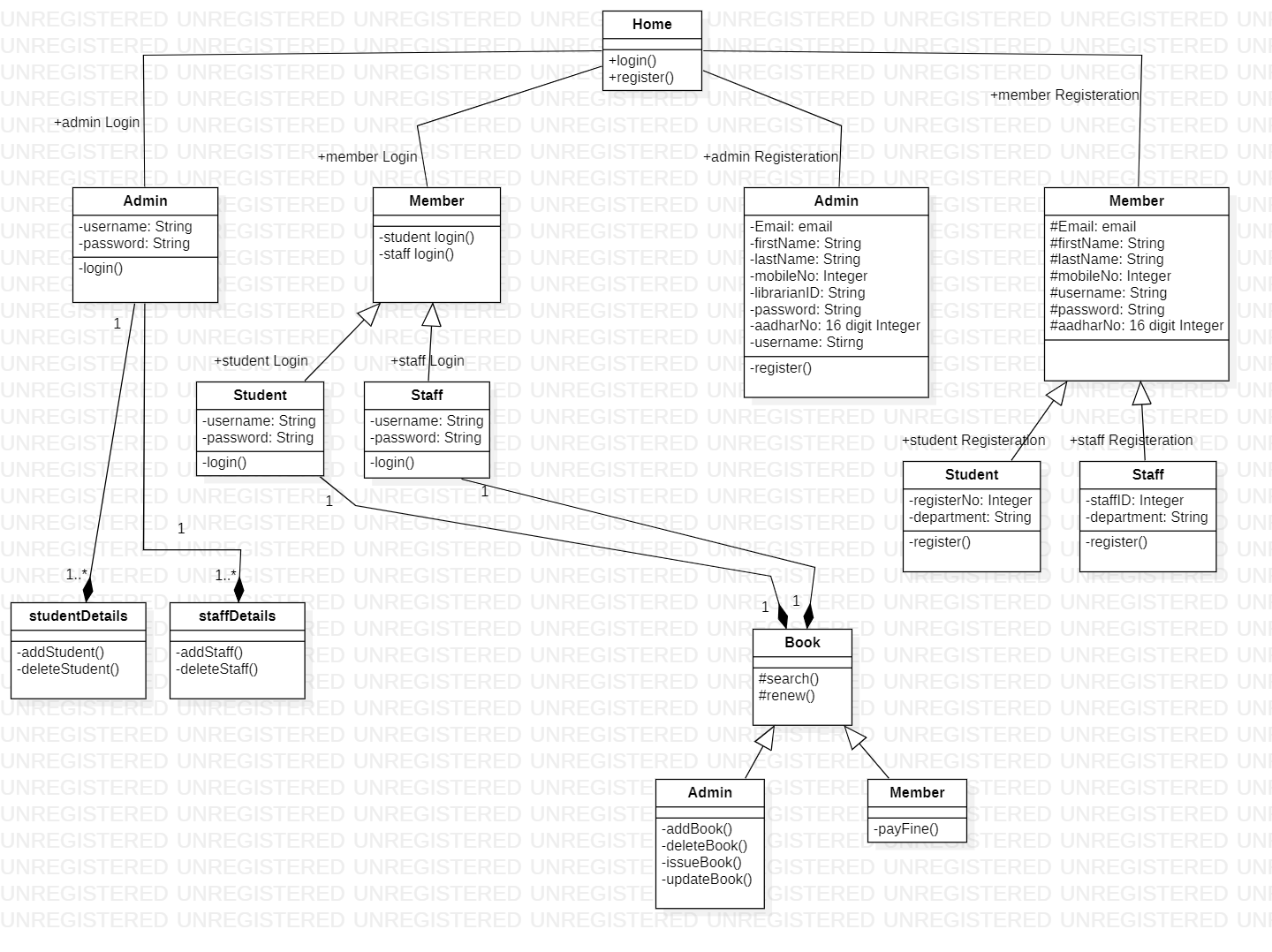
**6. Post conditions:**

List of books is updated according to added or removed books.

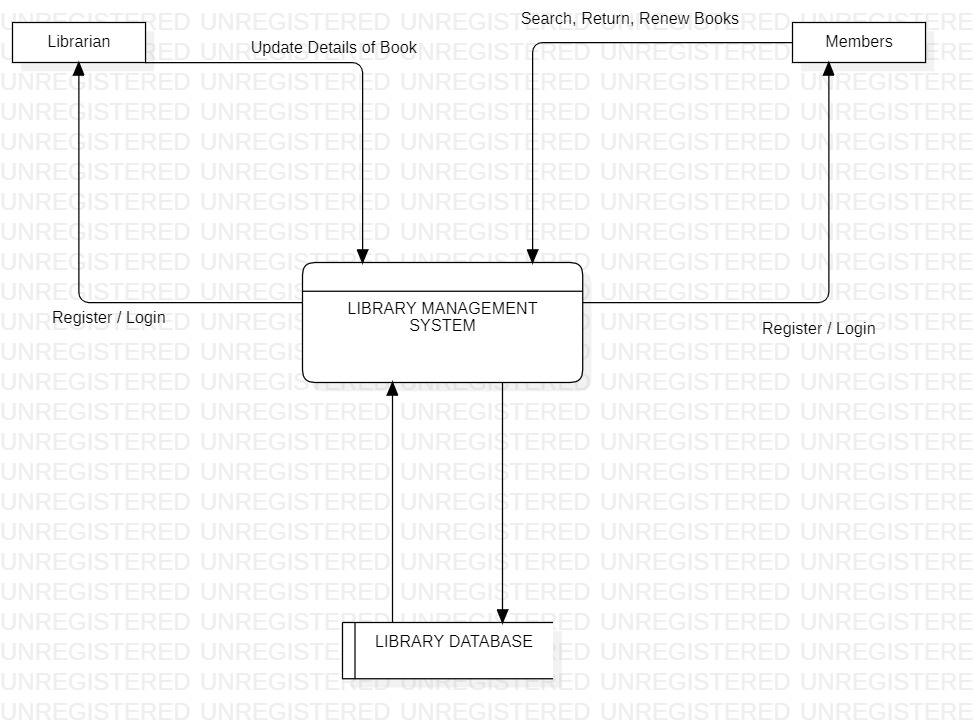
**7. Extension points:**

None.

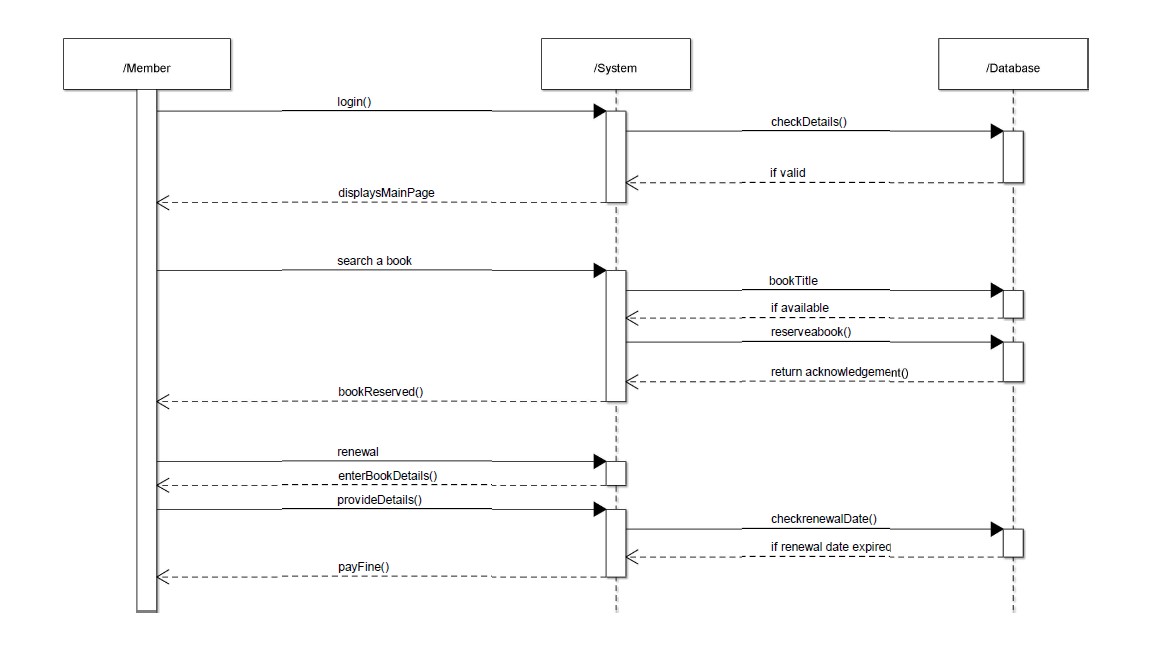
**9. CLASS DIAGRAM**

****

**10. DFD DIAGRAM**

****

**11. SEQUENCE DIAGRAM**

****

**12. Data Design**

**12.1 STUDENTS:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Data Item** | **Type** | **Length** | **Description** |
| 1 | STUDENTS | | INTEGER | STRING |
| 1.1 | First Name | String | 45 | First Name of the Student |
| 1.2 | Last Name | String | 45 | Last Name of the Student |
| 1.3 | Email | String | 45 | Email of the Student |
| 1.4 | Mobile No | String | 10 | Mobile No of the Student |
| 1.5 | Aadhar No | String | 16 | Aadhar No of the Student |
| 1.6 | Register No | String | 20 | Register No of the Student |
| 1.7 | Gender | String | 10 | Gender of the Student |
| 1.8 | Department | String | 10 | Department of the Student |
| 1.9 | Year | String | 10 | Year of the Student |
| 1.10 | DOB | Date | 10 | Date of the Birth of the Student |
| 1.11 | Username | String | 8 | Username of the Student |
| 1.12 | Password | String | 16 | Password of the Student |
| 1.13 | Count | INT | 10 | Limit of the Borrow the Book |

**12.2 STAFFS:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Data Item** | **Type** | **Length** | **Description** |
| 1 | STAFFS | | INTEGER | STRING |
| 1.1 | First Name | String | 45 | First Name of the Staff |
| 1.2 | Last Name | String | 45 | Last Name of the Staff |
| 1.3 | Email | String | 45 | Email of the Staff |
| 1.4 | Mobile No | String | 10 | Mobile No of the Staff |
| 1.5 | Aadhar No | String | 16 | Aadhar No of the Staff |
| 1.6 | Register No | String | 20 | Register No of the Staff |
| 1.7 | Gender | String | 10 | Gender of the Staff |
| 1.8 | Department | String | 10 | Department of the Staff |
| 1.9 | DOB | Date | 10 | Date of the Birth of the Staff |
| 1.10 | Username | String | 8 | Username of the Staff |
| 1.11 | Password | String | 16 | Password of the Staff |
| 1.12 | Count | INT | 10 | Limit of the Borrow the Book |

**12.3 LIBRARIAN:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Data Item** | **Type** | **Length** | **Description** |
| 1 | LIBRARIAN | | INTEGER | STRING |
| 1.1 | First Name | String | 45 | First Name of the Librarian |
| 1.2 | Last Name | String | 45 | Last Name of the Librarian |
| 1.3 | Email | String | 45 | Email of the Librarian |
| 1.4 | Mobile No | String | 10 | Mobile No of the Librarian |
| 1.5 | Aadhar No | String | 16 | Aadhar No of the Librarian |
| 1.6 | Librarian ID | String | 20 | Librarian ID of the Librarian |
| 1.7 | Gender | String | 10 | Gender of the Librarian |
| 1.8 | DOB | Date | 10 | Date of the Birth of the Librarian |
| 1.9 | Username | String | 8 | Username of the Librarian |
| 1.10 | Password | String | 16 | Password of the Librarian |

**12.4 BOOKS:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Data Item** | **Type** | **Length** | **Description** |
| 1 | BOOKS | | INTEGER | STRING |
| 1.1 | Book Name | String | 45 | Name of the Book |
| 1.2 | Author Name | String | 45 | Author Name of the Book |
| 1.3 | Email | String | 45 | Email of the Book |
| 1.4 | Mobile No | String | 10 | Mobile No of the Book |
| 1.5 | ISBN No | String | 16 | ISBN No of the Book |
| 1.6 | Published Year | String | 20 | Published Year of the Book |
| 1.7 | Publication | String | 10 | Publication of the Book |
| 1.8 | Edition | String | 10 | Edition of the Book |
| 1.9 | Count | INT | 10 | Count of the Book |

**THANK YOU**