5.6.3 Case Study of Library Management System

The problem statement for library management system is given below:

Problem statement

A software has to be developed for automating the manual library of a University. The system should be stand alone in nature. It should be designed to provide functionality's as explained below:

Issue of books

- A student of any course should be able to get books issued.
- Books from general section are issued to all but book bank books are issued only for their respective courses.
- A limitation is imposed on the number of books a student can issue.
- A maximum of 4 books from book bank and 3 books from general section is issued for 15 days only.
- The software takes the current system date as the date of issue and calculates date of return.
- A bar code detector is used to save the student as well as book information.
- The due date for return of the book is stamped on the book.

Return of books

- Any person can return the issued books.
- The student information is displayed using the bar code detector.
- The system displays the student details on whose name the books were issued as well as the date of issue and return of the book.
- The system operator verifies the duration for the issue.
- The information is saved and the corresponding updating take place in the database.

Query processing

The system should be able to provide information like:

- Availability of a particular book.
- Availability of book of any particular author.
- Number of copies available of the desired book.

The system should also be able to generate reports regarding the details of the books available in the library at any given time. The corresponding printouts for each entry (issue/return) made in the system should be generated. Security provisions like the 'login authenticity should be provided. Each user should have a user id and a password. Record of the users of the system should be kept in the log file. Provision should be made for full backup of the system.

Use cases

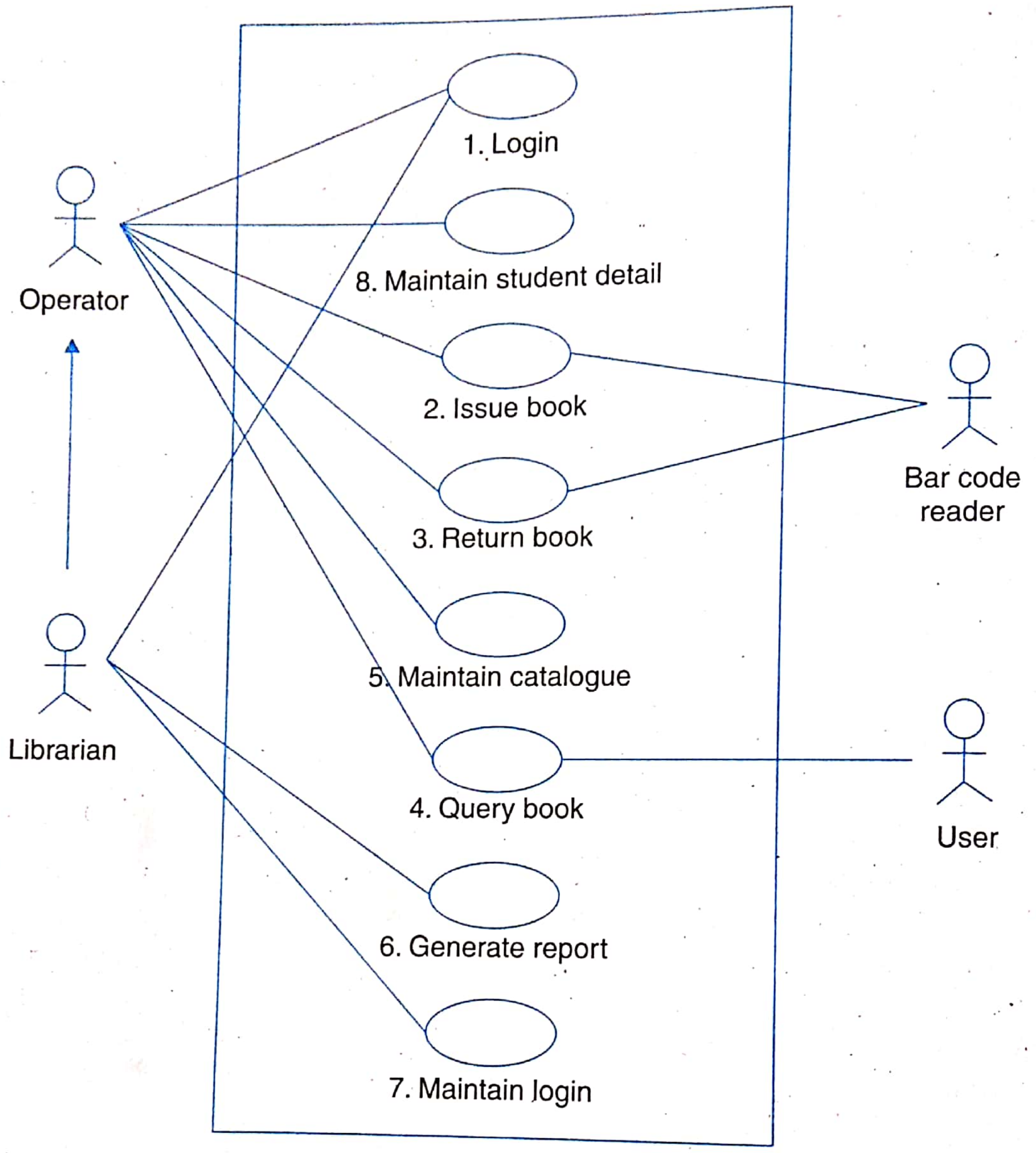
From the problem description, we can see that the system has four actors.

1. Librarian
2. Operator
3. Bar Code Reader
4. User

To define system's functionalities, we can view the system as a collection of following use cases:

1. Login
2. Issue Book
3. Return Book
4. Query Book

5. Maintain Catalog
6. Generate Report
7. Maintain Login
8. Maintain student Details



Use case diagram for library management system

USE CASE DESCRIPTION

1. LOGIN

1.1. Introduction

This use case documents the procedure for logging into the Library Management System based on user privileges.

- Operator (Issue Book, Return Book, Query a Book, Maintain Catalogue, Maintain Student Detail)
- Librarian (Generate Reports, Maintain Login)

1.2 Actors

Operator, Librarian

1.3 Pre-Condition

None

1.4 Post-Condition

If use case is successful, the user is logged into the system, otherwise the system state is unchanged.

1.5 Flow of Events

1.5.1 Basic Flow

This use case starts when actor wishes to log in to the Library Management System.

- The system requests that the actor enters his/her user_id and password.
- The actor enters user_id and password.
- The system validates the user_id and password and checks for his/her privileges.
- If the user is "operator", he/she will be logged into the system and presented with operator's menu.
- Otherwise, if the user is "librarian", he/she will be logged into the system and presented with librarian's menu.
- The use case ends.

1.5.2 Alternate Flow

1.5.2.1 Invalid Name/Password

If the system receives an invalid user_id or password, and error message is displayed and the use case ends.

1.6 Special Requirements

None

1.7 Related Use Cases

None

2. ISSUE BOOK

2.1 Introduction

This use case documents the procedure of issuing a book for following accounts:

- General (for 15 days)
- Book Bank (for the semester)

2.2 Actors

Operator, Barcode reader

2.3 Pre-Condition

Operator must be logged in to the system

2.4 Post-Condition

If use case is successful, the book is issued to the student in his/her general or book bank account, otherwise the system state is unchanged.

2.5 Flow of Events

2.5.1 Basic Flow

The use case starts when a student wants to get a book issued.

- The system reads and validates the student's information using the Bar Code Reader.
- The system reads the book's information using the Bar Code Reader.
- The return date of the book is calculated as per the account in which the student wishes to get the book issued – 15 days for General account and whole semester for Book Bank account.
- The book and student's information is saved into the database.
- The issue details are sent to the printer to generate the receipt.
- The use case ends.

2.5.2 *Alternate Flow*

2.5.2.1 Unauthorized Student

If the system doesn't validates the student, then an error message is flagged and the use case ends.

2.5.2.2 General Account is Full

If the student has requested a book in General Account and the later is full, *i.e.* he has already 3 books issued on his name, then the request for issue is denied and the use case ends.

2.5.2.3 Book Bank Account is Full

If the student has requested a book in Book Bank Account and the later is full, *i.e.* he has already 4 books issued on his name, then and error message is shown and the use case ends.

2.5.2.4 Course Mismatch between Student and the Book Bank Book.

If the student of a particular course has requested a book from some other course on Book Bank Account, the an option for getting the book issued in general account is given and the use case ends.

2.6 Special Requirements

None

2.7 Related Use Cases

Generate Barcode

3. RETURN BOOK

3.1 Introduction

This use case documents the procedure of returning a book and calculating the fine amount if the students has returned the book after the specified return date.

3.2 Actors

Operator, Barcode Reader

3.3 Pre-Condition

Operator must be logged in to the system.

3.4 Post-Condition

If use case is successful, the book is returned back to the library and if needed, the fine is calculated, otherwise the system state is unchanged.

3.5 Flow of Events

3.5.1 Basic Flow

This use case starts when a student wants to return a book.

- The system reads the book's information using the Bar Code Reader.
- The book is returned to the library.
- The database entries corresponding both to the student account and the book are updated.
- The return details are sent to the printer to generate the receipt.
- The use case ends.

3.5.2 Alternate Flow

3.5.2.1 Late return of book

If the book is returned after the due date, fine is calculated and database is updated accordingly. The use case end here.

3.6 Special Requirements

None

3.7 Related Use Cases

Generate Barcode

4. QUERY A BOOK

4.1 Introduction

This use case documents the procedure for searching a book based on the specified criteria, which are:

- Search by Author Name
- Search by Title Name

4.2 Actors

Operator, user

4.3 Pre-Condition

Operator user must be logged in to the system

4.4 Post-Condition

If use case is successful, the book details are displayed.

4.5 Flow of Events

4.5.1 Basic Flow

This use case starts when a student wants to search for a particular book.

- The system displays the various search criteria to the user.
- The user selects the search criteria.
- The result is displayed to the user.
- The use case ends.

4.6 Special Requirements

None

4.7 Related Use Cases

None

5. MAINTAIN CATALOG

5.1 Introduction

This use case documents the procedure for updating the catalog of the library.

5.2 Actors

Operator

5.3 Pre-Condition

Operator must be logged into the system.

5.4 Post-Condition

If use case is successful, the book should be updated, otherwise the system state in unchanged.

5.5 Flow of Events

5.5.1 Basic Flow

This use case starts when the operator wishes to add, delete or modify some details in the library.

- The corresponding changes are saved in the database.
- The use case ends.

5.5.2 Alternate Flow

None

5.6 Special Requirements

None

5.7 Related Use Cases

None

6. GENERAL REPORTS

6.1 Introduction

This use case documents the procedure for generating the reports as desired by the Librarian.

6.2 Actors

Librarian

6.3 Pre-Condition

Librarian must be logged into the system.

6.4 Post-Condition

If use case is successful, the various reports, regarding the details of the books available in the library at any given time, are generated.

6.5 Flow of Events

6.5.1 Basic Flow

This use case starts when a librarian wants to generate reports of the books available in the library.

- The system displays the various report generating criteria to the user, which can be the books issued to the students at a particular time, books available in the library etc.

- The librarian selects the criteria and enters the various parameters based on the criteria selected.
- The system generates the report and sends that to printer.
- The use case ends.

6.5.2 Alternate Flow

6.5.2.1 Printer out of paper or low on ink

If the printer goes out of paper or low on ink, then the printing operation is aborted and the necessary action needs to be taken, which can be feeding paper to the printer or replacing the ink cartridge. The use case ends.

6.6 Special Requirements

None

6.7 Related Use Cases

None

7. MAINTAIN LOGIN

7.1 Introduction

This use case documents the procedure for maintaining Login Details.

7.2 Actors

Librarian.

7.3 Pre-Condition

Librarian must be logged into the system.

7.4 Post-Condition

If use case is successful, the Login details should be updated, otherwise the system state is unchanged.

7.5 Flow of Events

7.5.1 Basic Flow

This use case starts when the Librarian wishes to add, delete or modify some details of login.

- The corresponding changes will be done.
- The use case ends.

7.5.2 Alternate Flow

None

7.6 Special Requirements

None

7.7 Related Use Cases

None

8. MAINTAIN STUDENT DETAILS

8.1 Introduction

This use case documents the procedure for maintaining student details.

8.2 Actors

Operator

8.3 Pre-Condition

Operator must be logged into the system.

8.4 Post-Condition

If use care is successful, the student details should be updated, otherwise the system state is unchanged.

8.5 Flow of Events

8.5.1 Basic Flow

This use case starts when the operator wishes to add, delete or modify some details of student.

- The corresponding changes will be done.
- The use case ends.

8.5.2 Alternate Flow

None

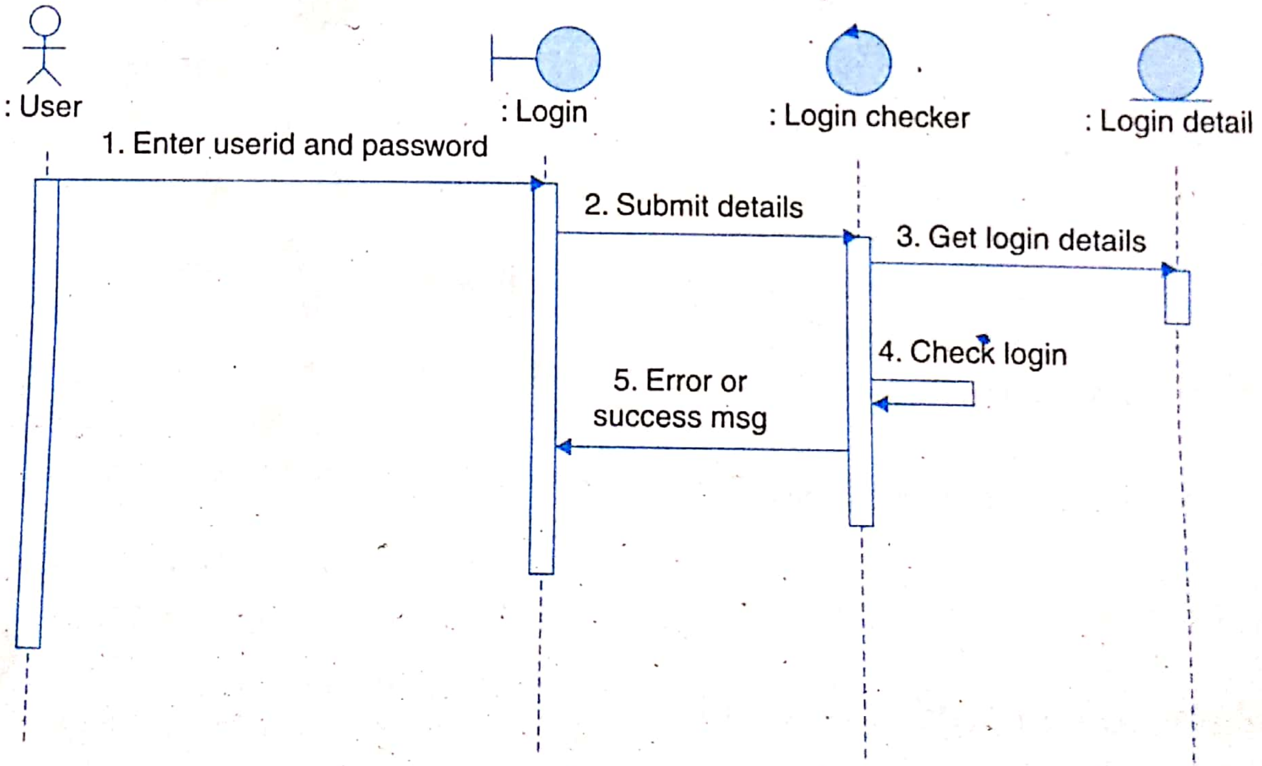
8.6 Special Requirements

None

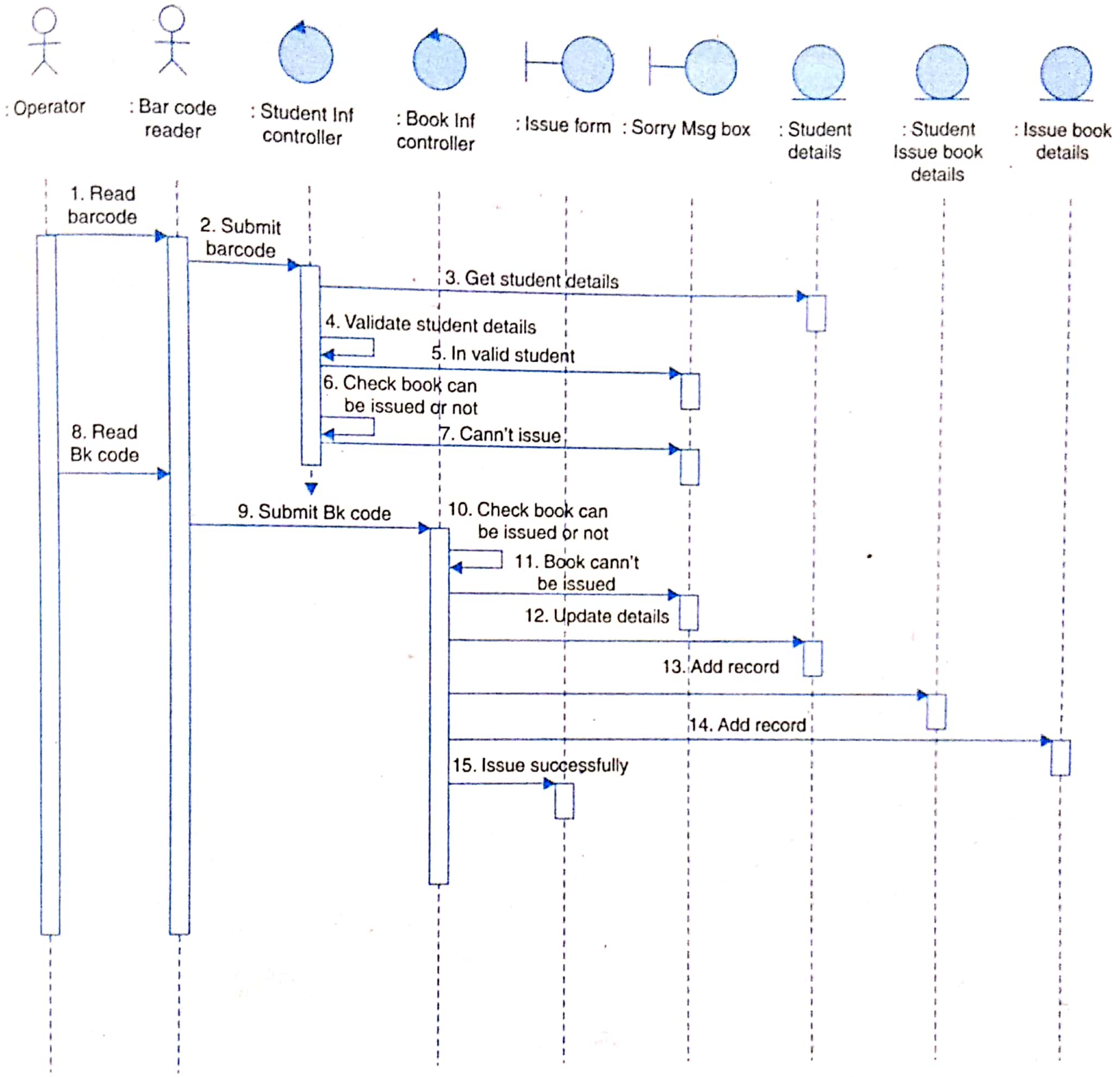
8.7 Related Use Cases

None

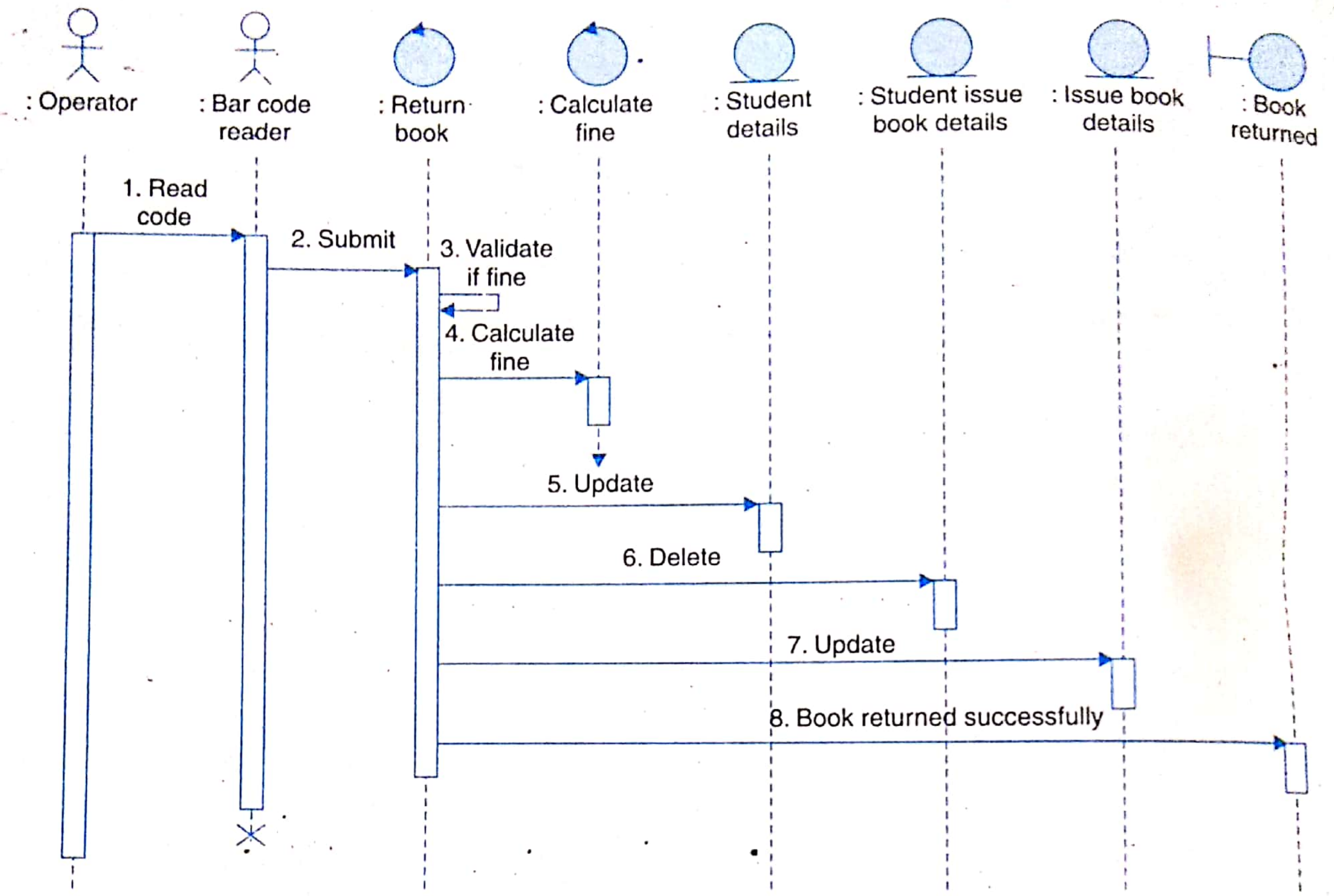
Sequence diagrams



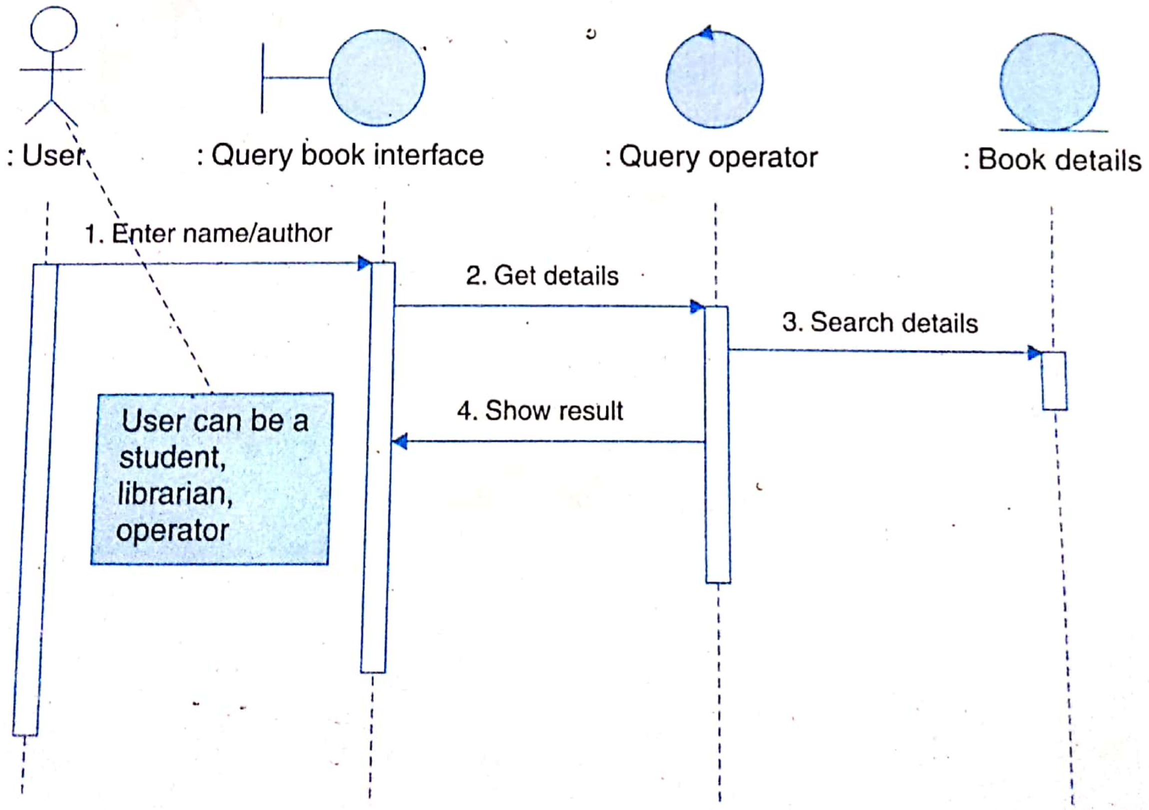
Sequence diagram—Login



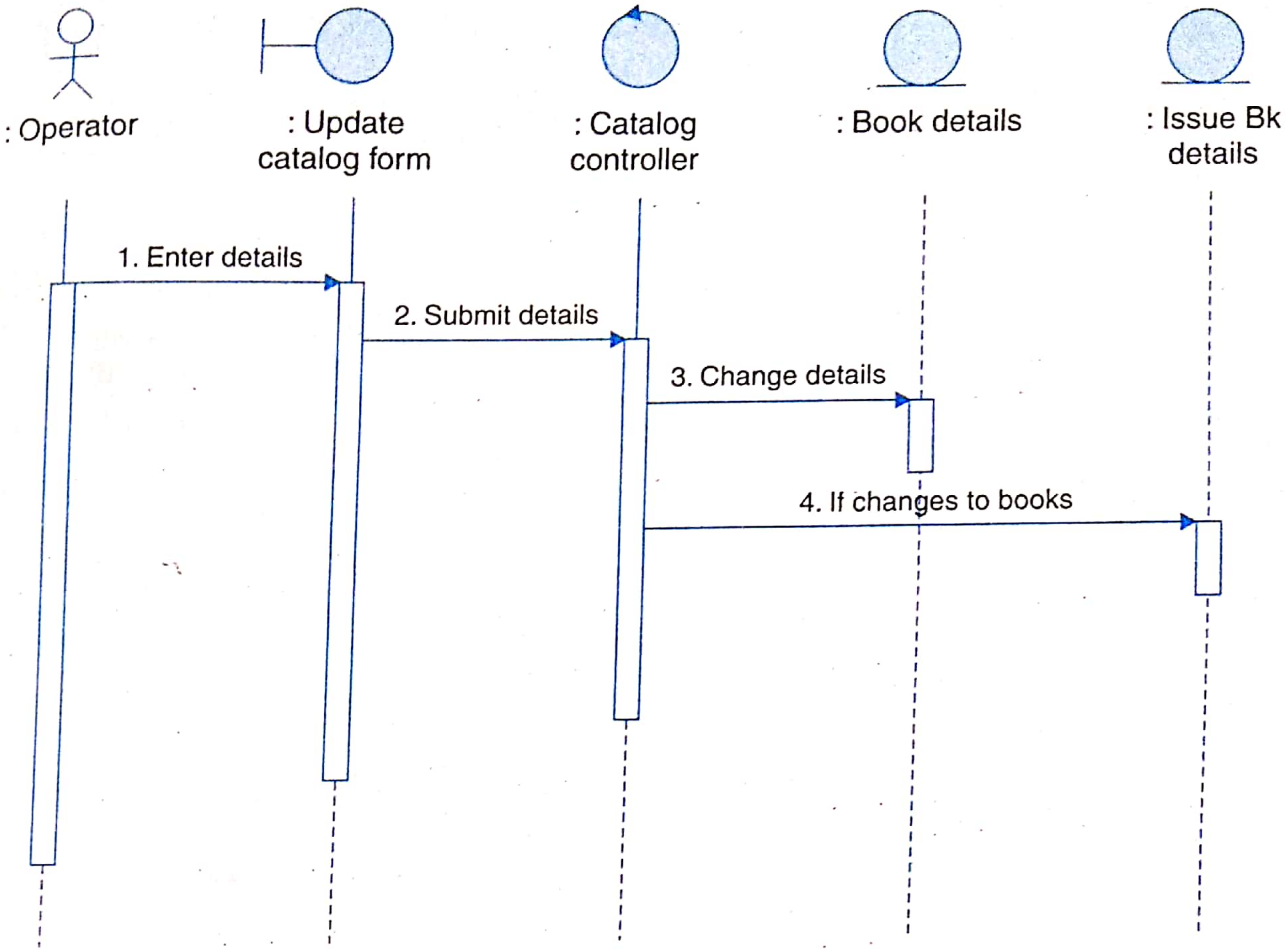
Sequence diagram—issue book



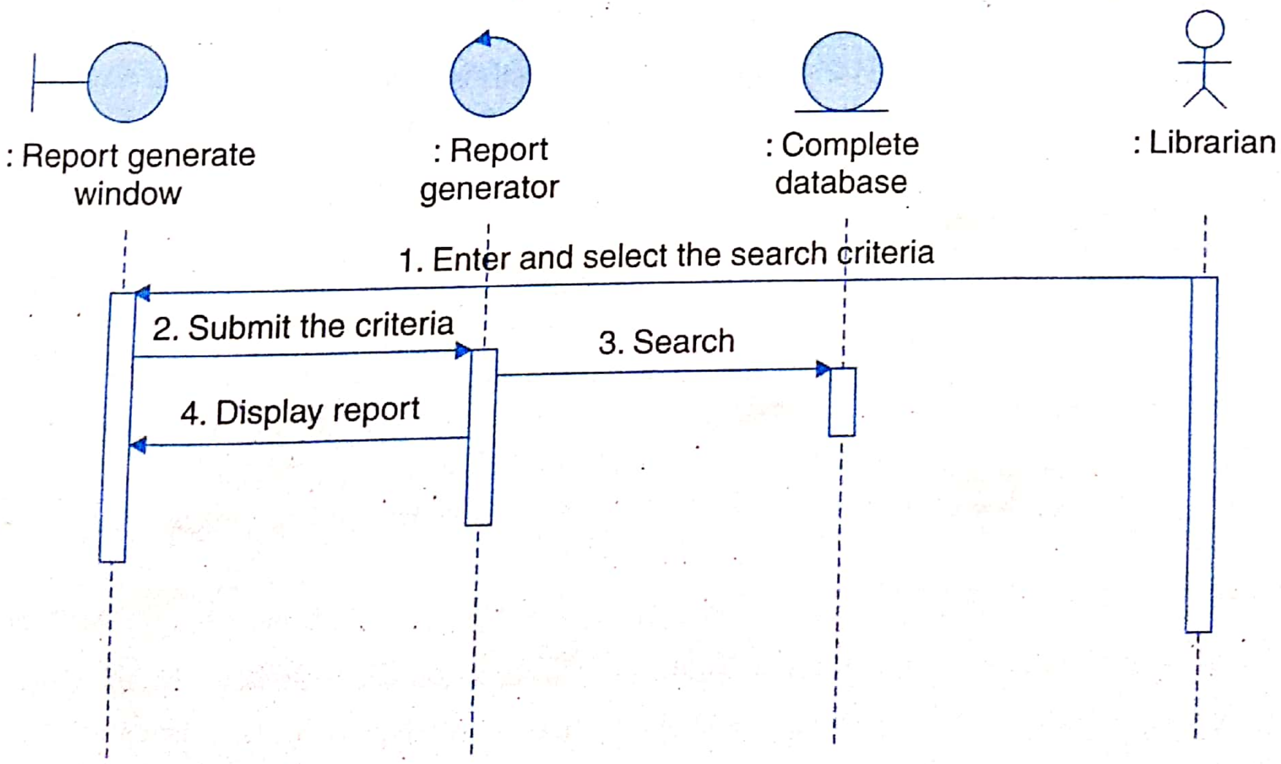
Sequence diagram—return book



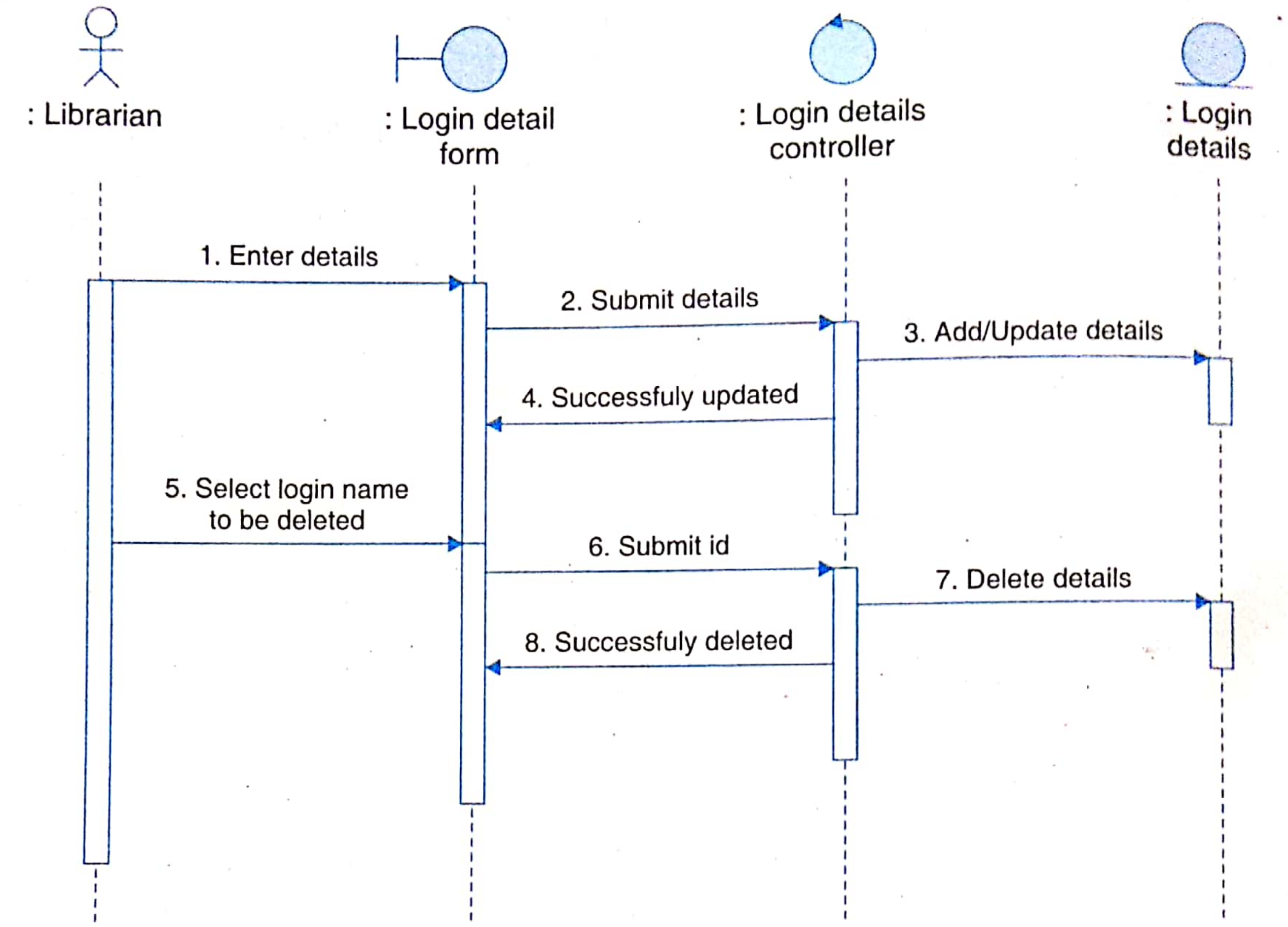
Sequence diagram—query book



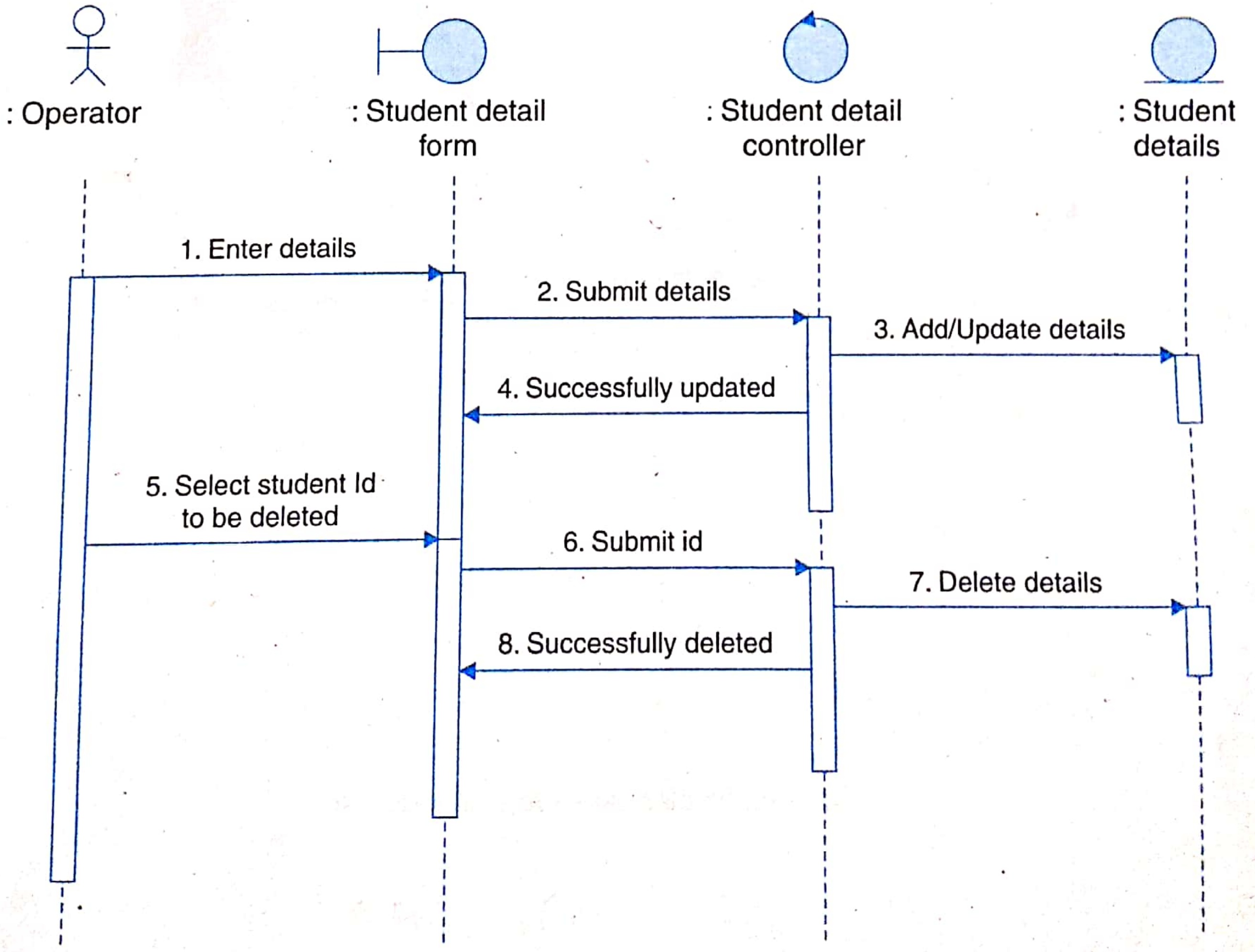
Sequence diagram—maintain catalog



Sequence diagram—generate reports

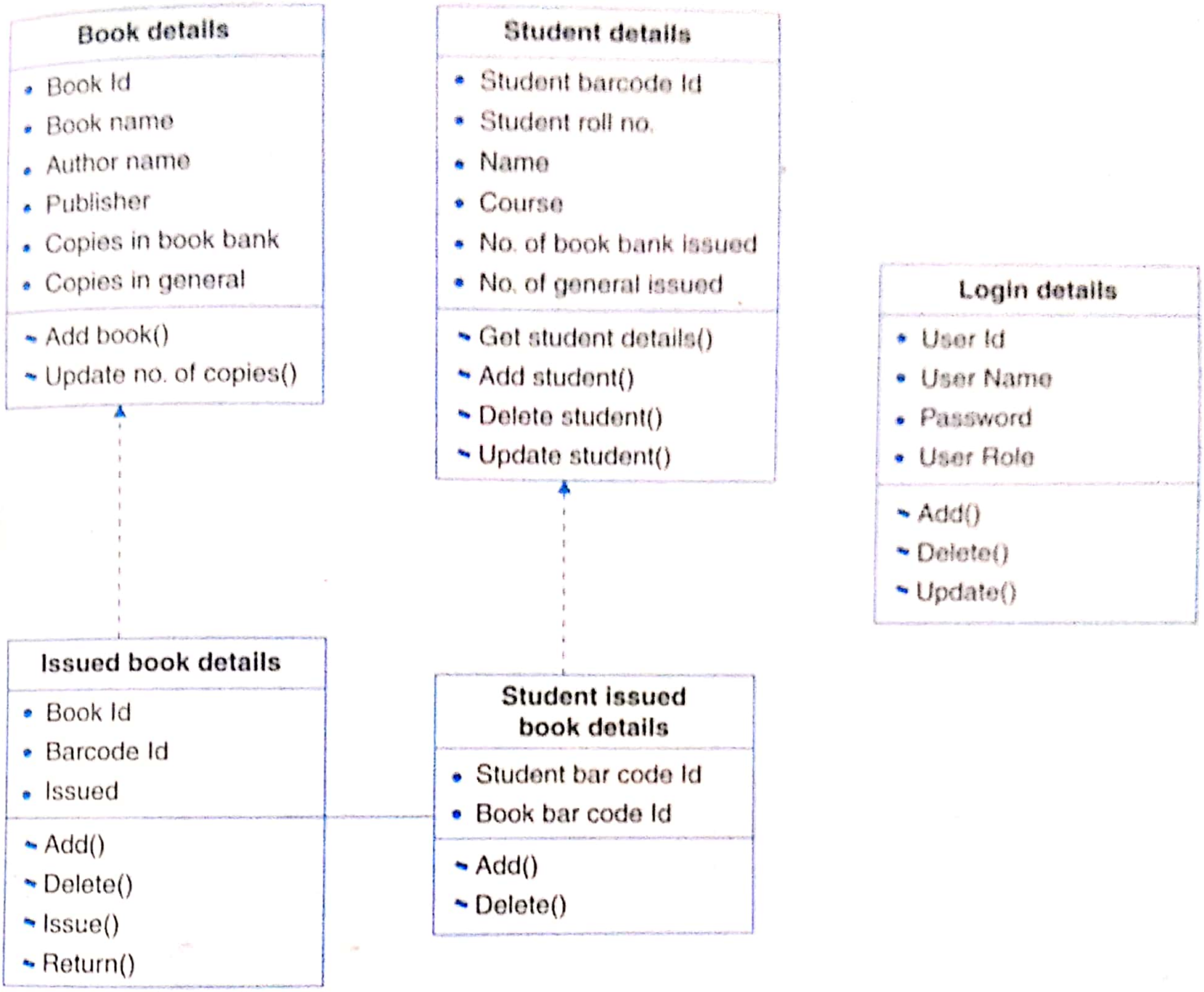


Sequence diagram—maintain login



Sequence diagram—maintain student details

Class diagram of entity classes



Class diagram of entity classes

In this design, we have shown class diagram of entity classes only. These entity classes generally becomes tables in the database. The other diagrams of the design document can be drawn using Rational Rose facilities (or other tools) and may or may not be required in each case study. All these diagrams are the foundations of the implementation phase.