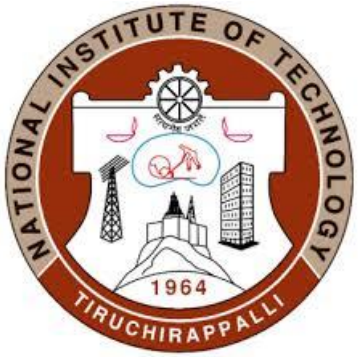
NATIONAL INSTITUTE OF TECHNOLOGY, TIRUCHIRAPPALLI-15



BATCH-2019-22

DBMS LAB

PROJECT ON

LIBRARY MANAGEMENT SYSTEM

SUBMITTED BY

SHILPI GUPTA
205119095
SECTION:A
MCA(I Year)

UNDER THE GUIDANCE OF
Prof. Dr. U. Vignesh

ACKNOWLEDGEMENT

Presentation, inspiration and motivation have always played a key role in the success of our venture.

I would like to express my sincere thanks and gratitude to my faculty ***Prof. Dr. U. Vignesh*** *to encourage me to the highest peak and to provide me the opportunity to prepare this project.*

I pay my deep sense of gratitude to ***Dr. P.J.A Alphonse, Head of the Department (Computer Applications)*** *who gave me the golden opportunity.*

Secondly I would also like to extend my sincere thanks to my parents who supported and motivated me a lot in finalizing this project within the limited time frame. So, with due regards, I express my gratitude to them.

Shilpi Gupta

205119095

MCA(I Year)

CONTENTS

ABSTRACT

Library management system is a project which aims in developing a computerized system to maintain all the daily work of library .This project has many features which are generally not available in normal library management systems. It also has a facility of admin login through which the admin can monitor the whole system.

It also has facility of an online notice board where teachers can student can put up information about workshops or seminars being held in our colleges or nearby colleges and librarian after proper verification from the concerned institution organizing the seminar can add it to the notice board. It has also a facility where student after logging in their accounts can see list of books issued and its issue date and return date and also the students can request the librarian to add new books by filling the book request form.

The librarian after logging into his account ie admin account can generate various reports such as student report , issue report, teacher report and book report Overall this project of ours is being developed to help the students as well as staff of library to maintain the library in the best way possible and also reduce the human efforts.

INTRODUCTION

This section gives an overview about the aim, objectives, background and operation environment of the system.

PROJECT AIMS AND OBJECTIVES

The aims and objectives are as follows:

- Computerized book issue facility.
- Librarian can add new book any time he wants.
- A login page for the Librarian to ensure user authentication.
- Computerized book return facility by just entering Book ID and Student ID.
- A newly admitted student can be added any time with his details.
- A Librarian login page where Librarian can add any new book, new student, issue book, return book.
- Issue/return book details can be seen just by one click. Which student got any book issued all the details can be seen.

BACKGROUND OF PROJECT

Library Management System is an application which refers to library systems which are generally small or medium in size. It is used by librarian to manage the library using a computerized system where he/she can record various transactions like issue of books, return of books, addition of new books, addition of new students etc.

Books and student maintenance modules are also included in this system which would keep track of the students regarding issue/return books in the library and also a detailed description about the books a library contains. With this computerized system there will be no loss of book record or member record which generally happens when a non-computerized system is used. In addition, report module is also included in Library Management System.

The Librarian is able to generate different kinds of reports like lists of students registered, list of books, issue and return reports. All these modules are able to help librarian to manage the library with more convenience and in a more efficient way as compared to library systems which are not computerized.

RESOURCES USED

HARDWARE RESOURCES:

System Type	:64-bitOperating System
CPU Type	: Intel (R) Core(TM)
Main Processor	: i(5) 4440
CPU Clock Speed	: 3.10 GHz
Installed RAM	: 8 GB

SOFTWARE RESOURCES:

Operating System	: Windows 10
Front-end software	: NetBeans IDE 8.2
Back-end software	: MySQL Server 5.5
Word Processing Software	: MS-Word

SYSTEM ANALYSIS

In this section, we will discuss and analyze about the developing process of Library Management System including software requirement specification (SRS) and comparison between existing and proposed system. Besides that, existing vs proposed provides a view of how the proposed system will be more efficient than the existing one.

SOFTWARE REQUIREMENT SPECIFICATION

➢ PRODUCT DESCRIPTION:

Library Management System is a computerized system which helps user (librarian) to manage the library daily activity in electronic format. It reduces the risk of paper work such as file lost, file damaged and time consuming. It can help user to manage the transaction or record more effectively and timesaving.

➢ PROBLEM STATEMENT:

The problem occurred before having computerized system includes:

- ***File lost***
  When computerized system is not implemented file is always lost because of human environment. Sometimes due to some human error there may be a loss of records.

- ***File damaged***
  When a computerized system is not there file is always lost due to some accident like spilling of water by some member on file accidentally. Besides some natural disaster like floods or fires may also damage the files.

- ***Difficult to search record***

When there is no computerized system there is always a difficulty in searching of records if the records are large in number.

- ***Space consuming***
  After the number of records become large the space for physical storage of file and records also increases if no computerized system is implemented.

- ***Cost consuming***
  As there is no computerized system to add each record paper will be needed which will increase the cost for the management of library.

SYSTEM OBJECTIVES

- ***Improvement in control and performance***
  The system is developed to cope up with the current issues and problems of library .The system can add user, validate user and is also bug free.

- ***Save cost***
  After computerized system is implemented less human force will be required to maintain the library thus reducing the overall cost.

- ***Save time***
  Librarian is able to search record by using few clicks of mouse and few search keywords thus saving his/her valuable time.

SYSTEM REQUIREMENTS

- **EFFICIENCY REQUIREMENT**

  When library management system is implemented, librarian will easily access the library as searching and book transaction will be very faster .

- **RELIABILITY REQUIREMENT**

  The system accurately performs admin login, admin validation , report generation, book transaction and search.

- **USABILITY REQUIREMENT**

  The system is designed for a user friendly environment so that the librarian can perform the various tasks easily and in an effective way.

- **ORGANIZATIONAL IMPLEMENTATION REQUIREMNTS**

  In implementing whole system it uses NETBEANS IDE 8.2 in frontend and the backend that is the database part is developed using MySQL 5.5. And furthermore Connectivity is done between frontend and database.

SOFTWARE TOOLS USED

The whole Project is divided in two parts the front end and the back end.

FRONT END

The front end is designed using NETBEANS IDE 8.2

- **NETBEANS IDE 8.2**

  NetBeans IDE 8.2 provides out-of-the-box code analyzers and editors for working with the latest Java 8 technologies--Java SE 8, Java SE Embedded 8, and Java ME Embedded 8. The IDE also has a range of new tools for HTML5/JavaScript, in particular for Node.js, KnockoutJS, and AngularJS; enhancements that further improve its support for Maven and Java EE with PrimeFaces; and improvements to PHP and C/C++ support.

BACK END

The back end is designed using MYSQL which is used to design the database .

- **MY SQL 5.5:**

  ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements .MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

PROGRAM DESCRIPTION

PACKAGES IMPORTED

- import java.sql.*;
- import javax.swing.JFrame;
- import javax.swing.JOptionPane;
- import javax.swing.table.DefaultTableModel;

CLASS VARIABLES AND METHODS

➢ login1.java

Variables Used

- jTextField1-To input username.
- jPasswordField1-To input password.

Methods used

✓ private void jButton1ActionPerformed**:**To Login to the application.

✓ private void jButton2ActionPerformed**:**To close the form

➢ home.java

Methods used

- ✓ private void jButton1ActionPerformed**:** Move to the ‘addstudent form’ to add new student details.
- ✓ private void jButton2ActionPerformed**:**To move to ‘addbook’ form to add new book details.
- ✓ private void jButton3ActionPerformed**:**To move to ‘issuebook’ form to issue book to the students.
- ✓ private void jButton4ActionPerformed**:**To move to ‘returnbook’ form to return book.
- ✓ private void jButton5ActionPerformed**:**To move to ‘details1’ form to check the records of the books issued and returned by the students.
- ✓ private void jButton6ActionPerformed**:**To logout from the session.

➢ addstudent.java

Variables Used

- jTextField1-To input Student ID of the student.
- jTextField2-To input Name of the student.
- jTextField3-To input Father’s Name of the student.
- jComboBox1-To input the Course Name of the student.
- jComboBox2-To input Branch Name of the student.

Methods used

- ✓ private void jButton1ActionPerformed**:**To Save the details of the student.
- ✓ private void jButton2ActionPerformed**:**To close the form.

➢ addbook.java

Variables Used

- jTextField1-To input Book ID.
- jTextField2-To input Book Name.
- jTextField3-To input Book Publisher's Name.
- jTextField4-To input price of the book.
- jTextField5-To input Publishing year of the book.

Methods used

- ✓ private void jButton1ActionPerformed**:**To Save the details of the book added.
- ✓ private void jButton2ActionPerformed**:**To close the form.

➢ issuebook.java

This Frame is used to issue book to the students.

Variables Used

- jTextField1-To input Book ID to be issued to the student.
- jTextField2-To input student ID of the student to be issued book.
- jTextField3-To input the student Name to whom the book is issued.
- jDateChooser1-To input the Issue Date of the book.
- jDateChooser2-To input Due Date of the book.

Methods used

- ✓ private void jButton1ActionPerformed**:**To Issue the book to the student.
- ✓ private void jButton2ActionPerformed**:**To close the form.

➢ returnbook.java

This Frame is used to return book by the student to the library.

Variables Used

- jTextField1-To input Book ID of the issued book.
- jTextField2-To input student ID of the student to whom the book is issued.
- jTextField3 (uneditable)-To display the Issue Date of the Book after Clicking the 'Search' button.
- jTextField4 (uneditable)- To display the Due Date of the Book after Clicking the 'Search' button.

Methods used

- ✓ private void jButton1ActionPerformed**:**To Return the book back to the library.
- ✓ private void jButton2ActionPerformed**:**To close the form.
- ✓ private void jButton3ActionPerformed**:**To search for the book Issue date and due date issued by the student with given book ID and Student ID.

➢ Details1.java

This Frame consists of three Buttons that is, whatever user want he can select. This frame directs the user to the forms to give details about the books issued and books returned, all the details of the student who got book issued and returned.

Methods used

- ✓ private void jButton1ActionPerformed**:**To move to the frame where you can see the details regarding books issued by the students, student ID ,Name, Book Name and all the other information.
- ✓ private void jButton2ActionPerformed**:** To move to the frame where you can see the details regarding books returned by the students, student ID ,Name, Book Name and all the other information.
- ✓ private void jButton3ActionPerformed**:**To close the form.

➢ idetails.java

This frame is used to display the details of the books issued to the students as well as all the details of the students in the form of a table.

Variables Used

- jTable1-To display details of the books issued.

Methods used

- ✓ private void jButton1ActionPerformed**:**To close the form.

➢ rdetails.java

This frame is used to display the details of the books returned by the students as well as all the details of the students in the form of a table.

Variables Used

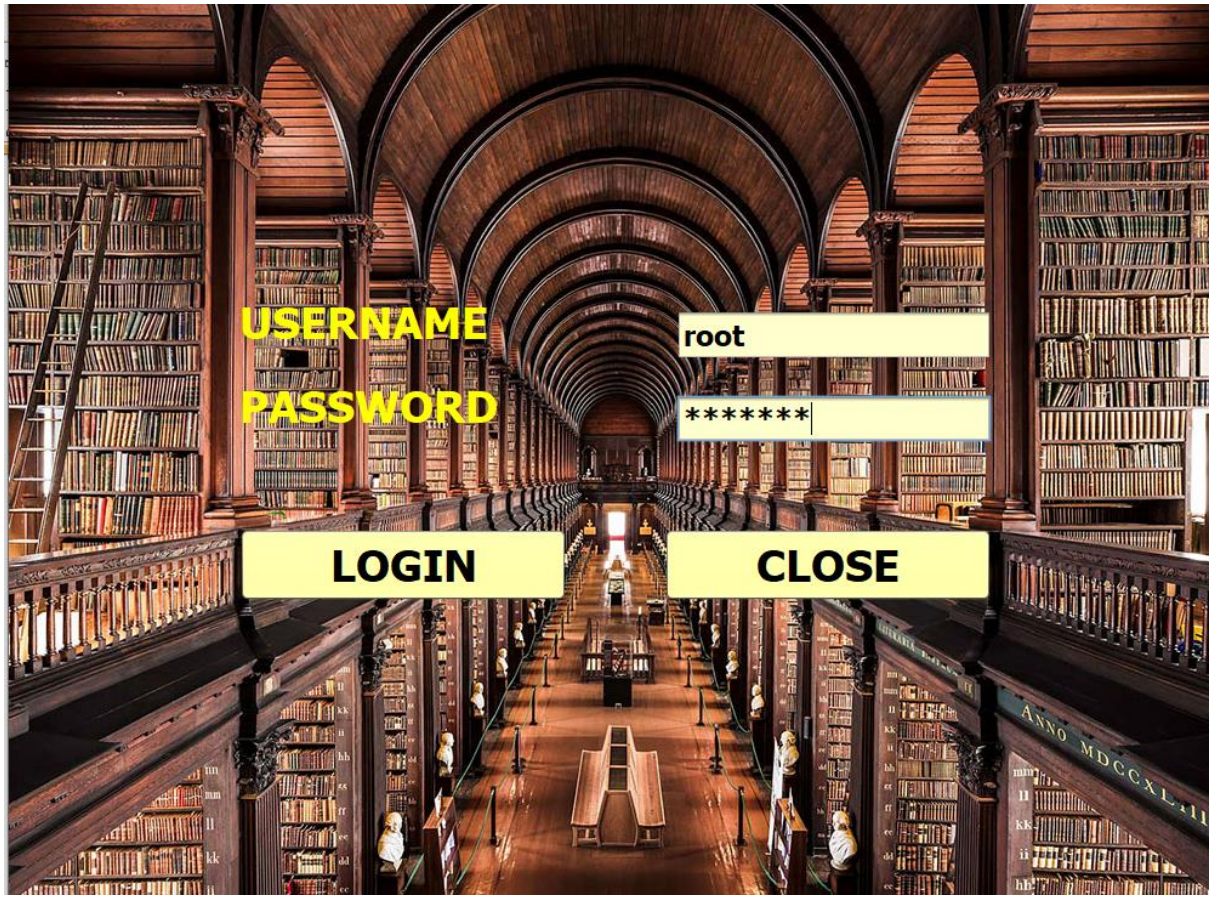
- jTable1-To display details of the books returned.

Methods used

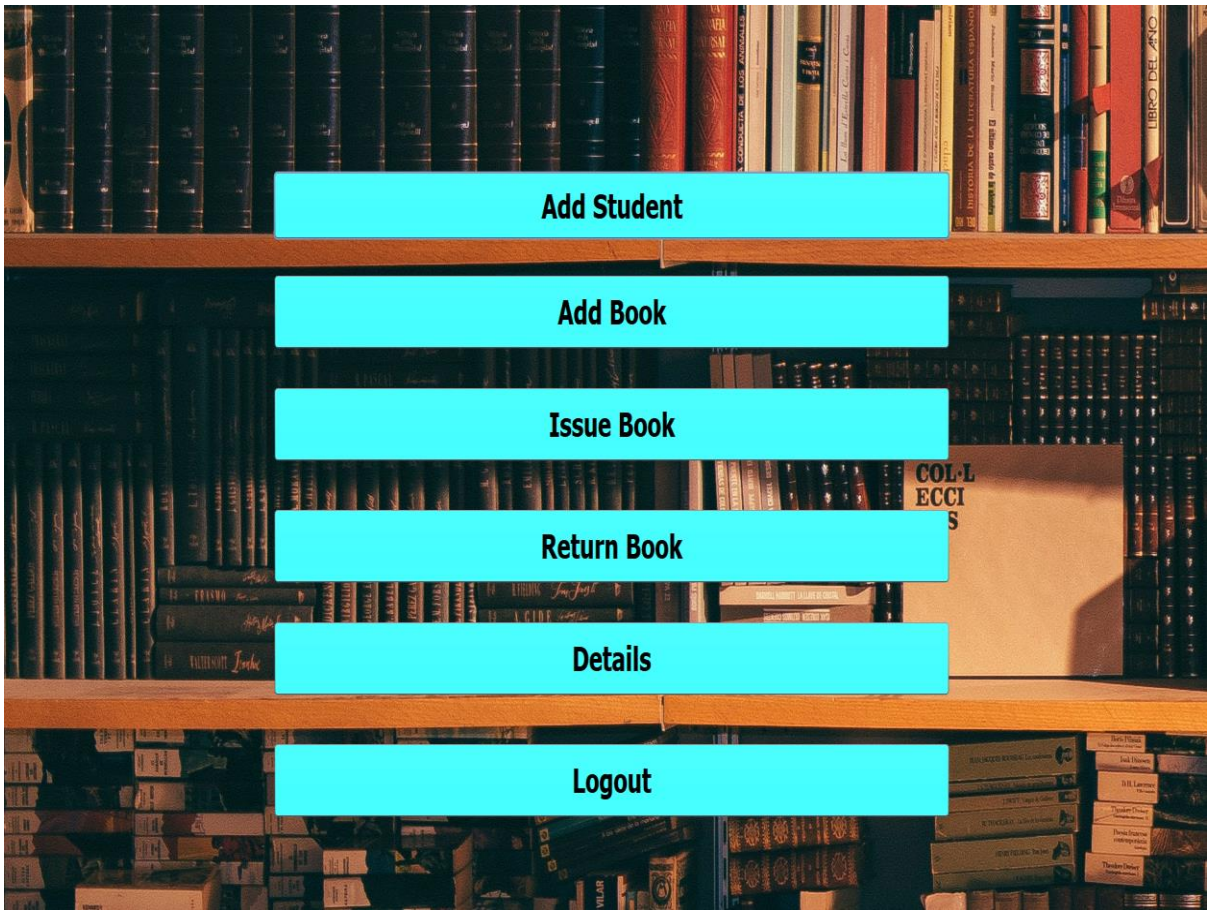
✓ private void jButton1ActionPerformed**:**To close the form.

SCREENSHOTS AND PICTURES

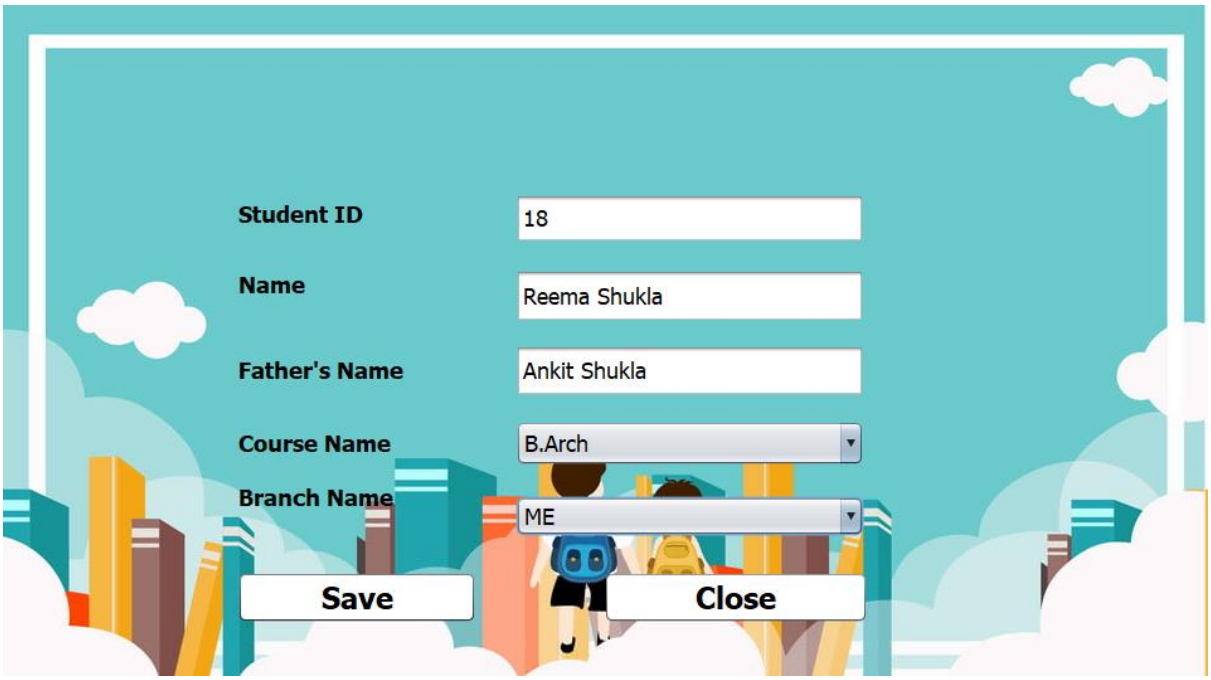
- ***Screenshot for Login Form***

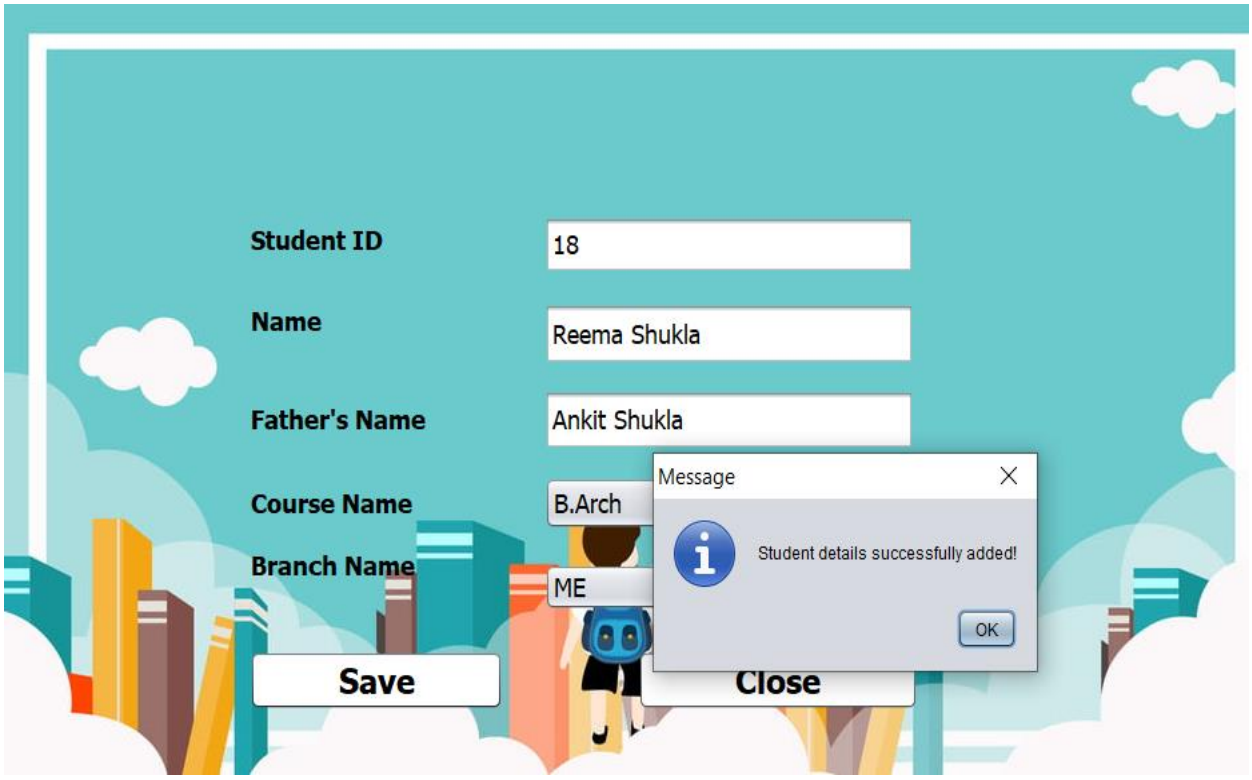


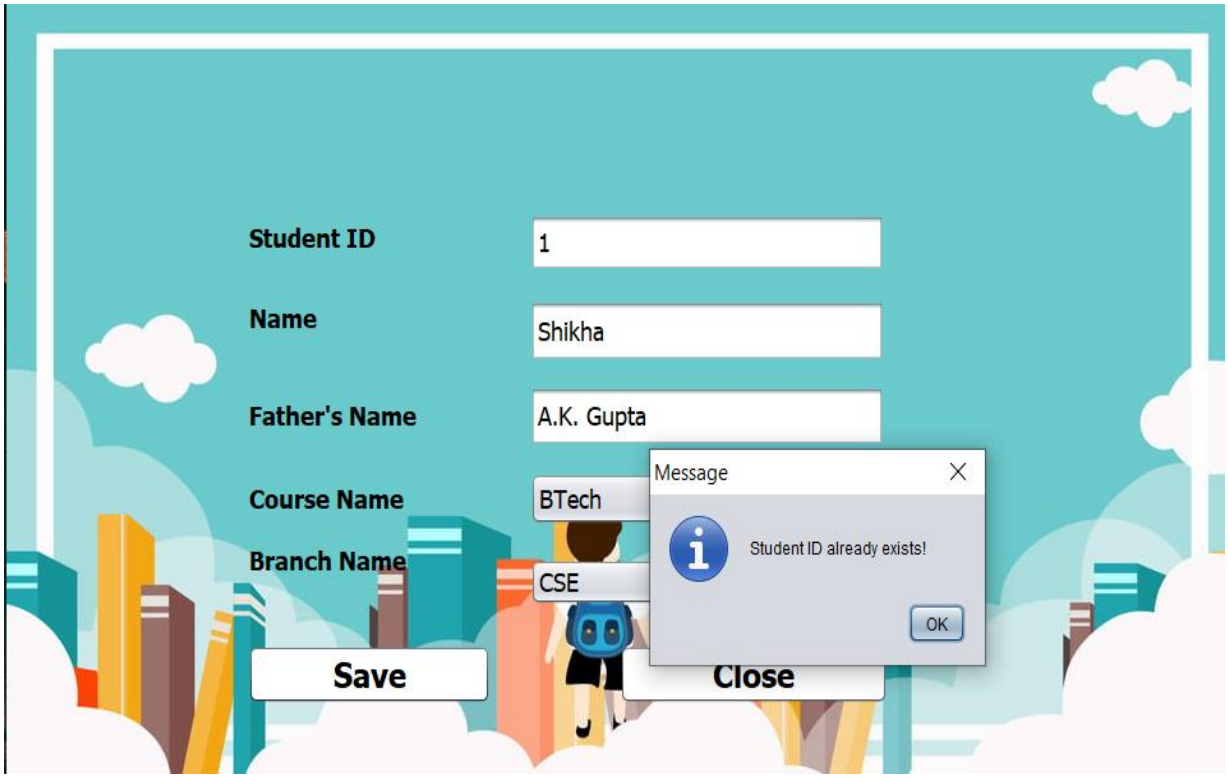
- ***Screenshot for Home Form***



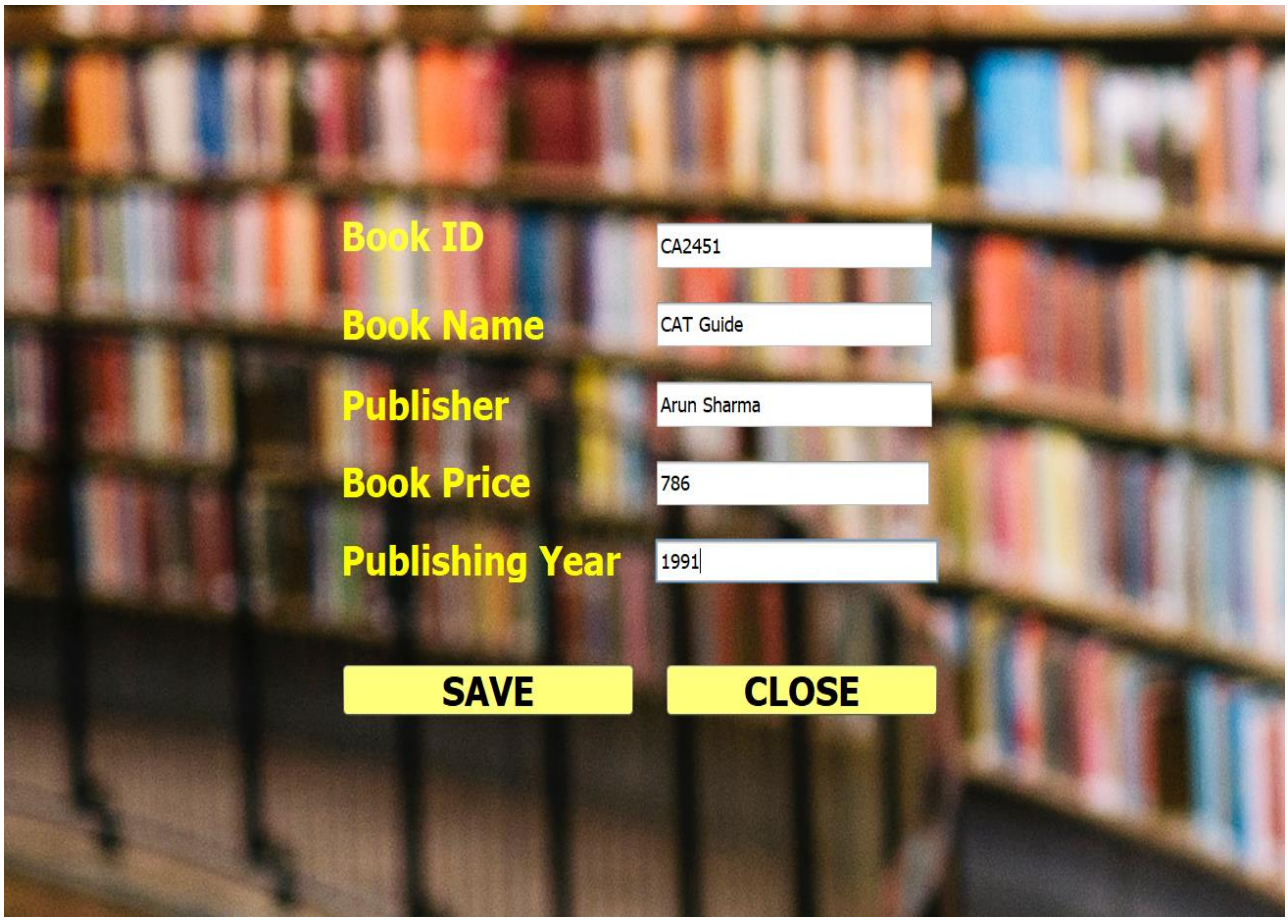
- ***Screenshot for Add Student Form***

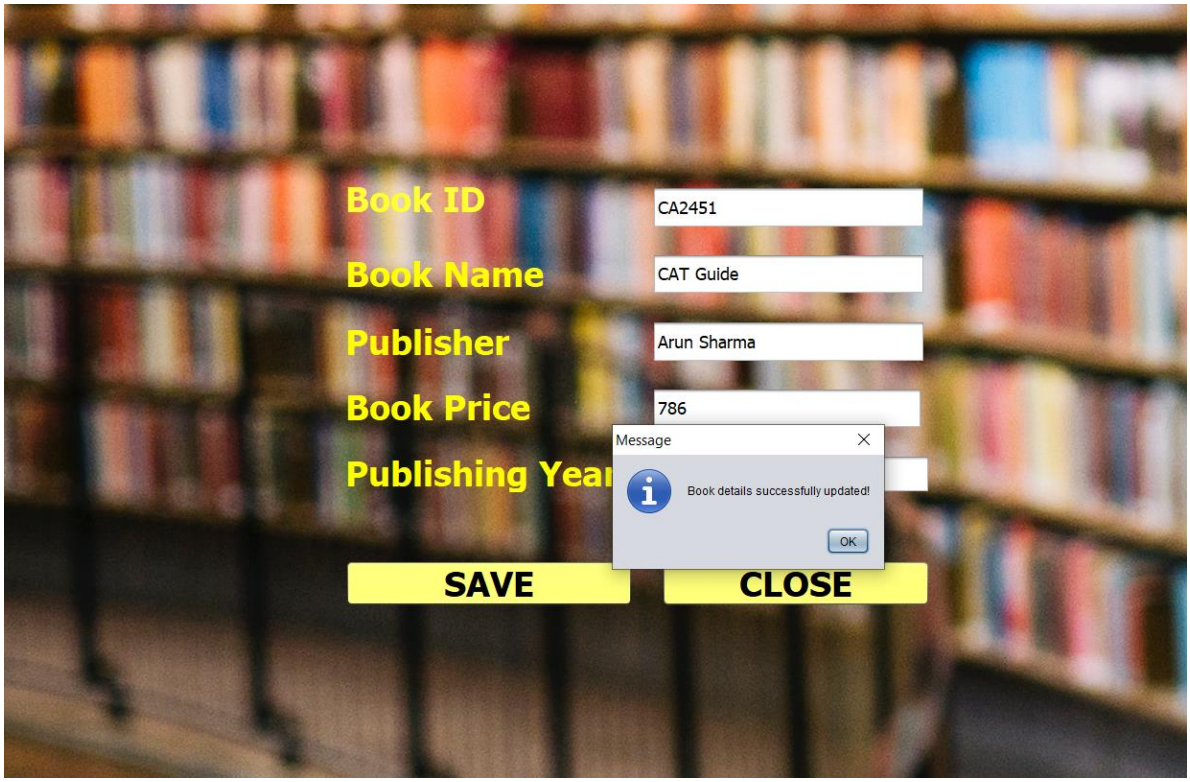





Student ID
1
Name
Shikha
Father's Name
A.K. Gupta
Course Name
BTech
Branch Name
CSE
Save
Close
Message
Student ID already exists!
OK

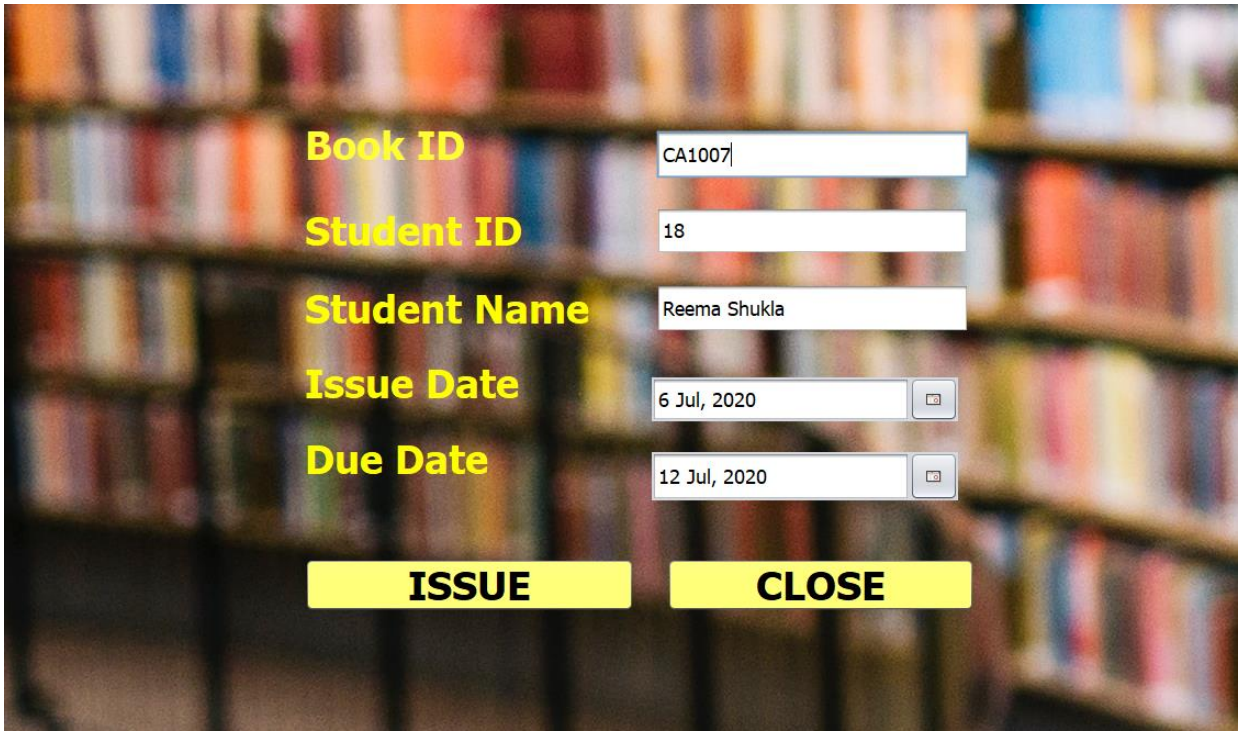
- ***Screenshot for Add Book Form***

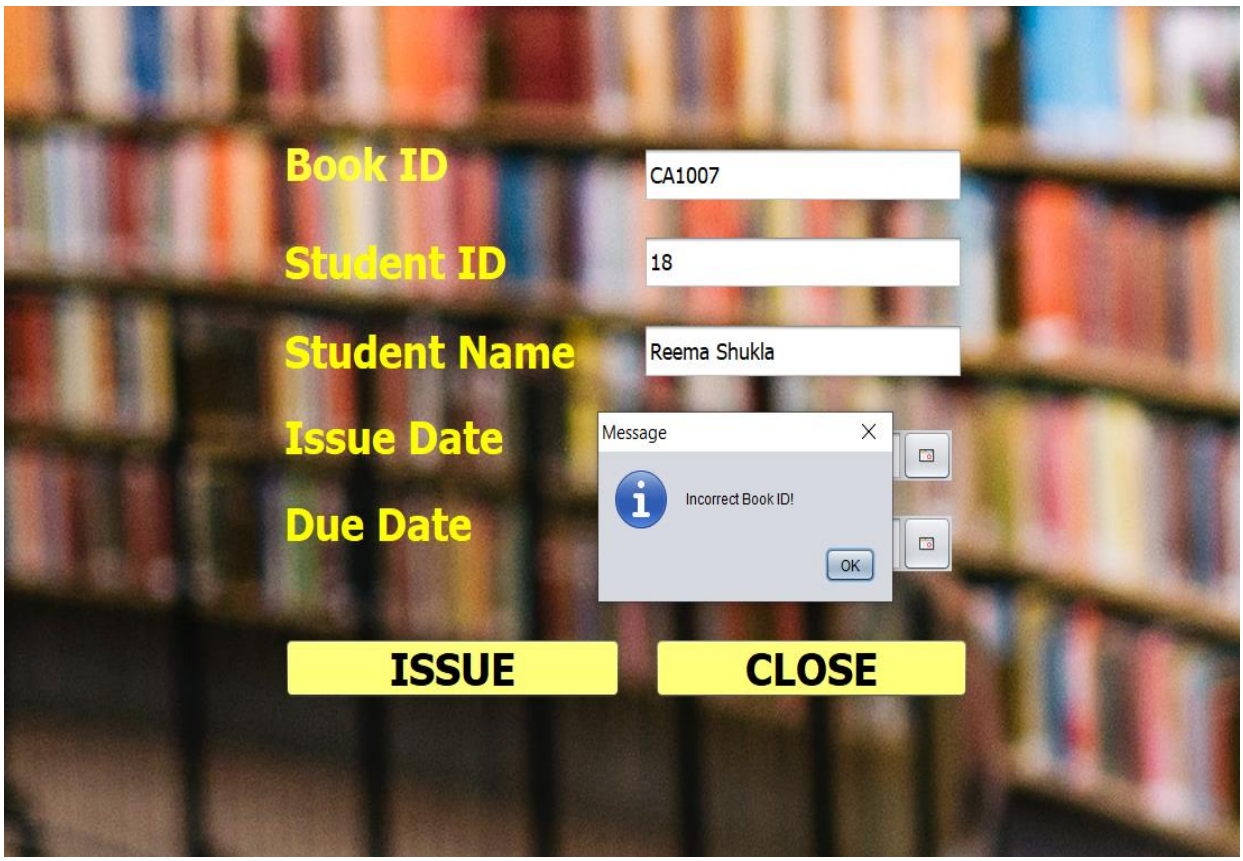


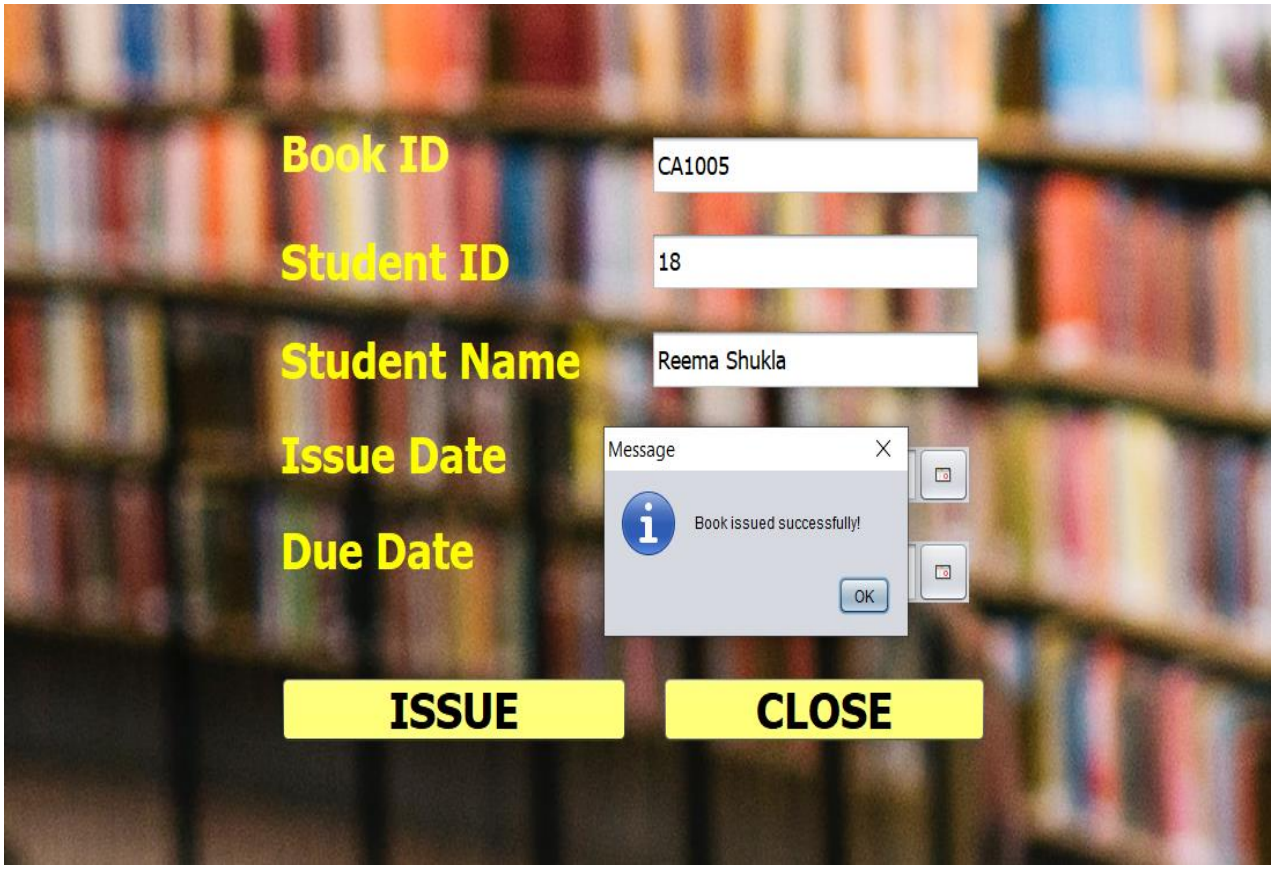

Book ID
CA2451
Book Name
CAT Guide
Publisher
Arun Sharma
Book Price
786
Publishing Year
Message
Book details successfully updated!
OK
SAVE
CLOSE

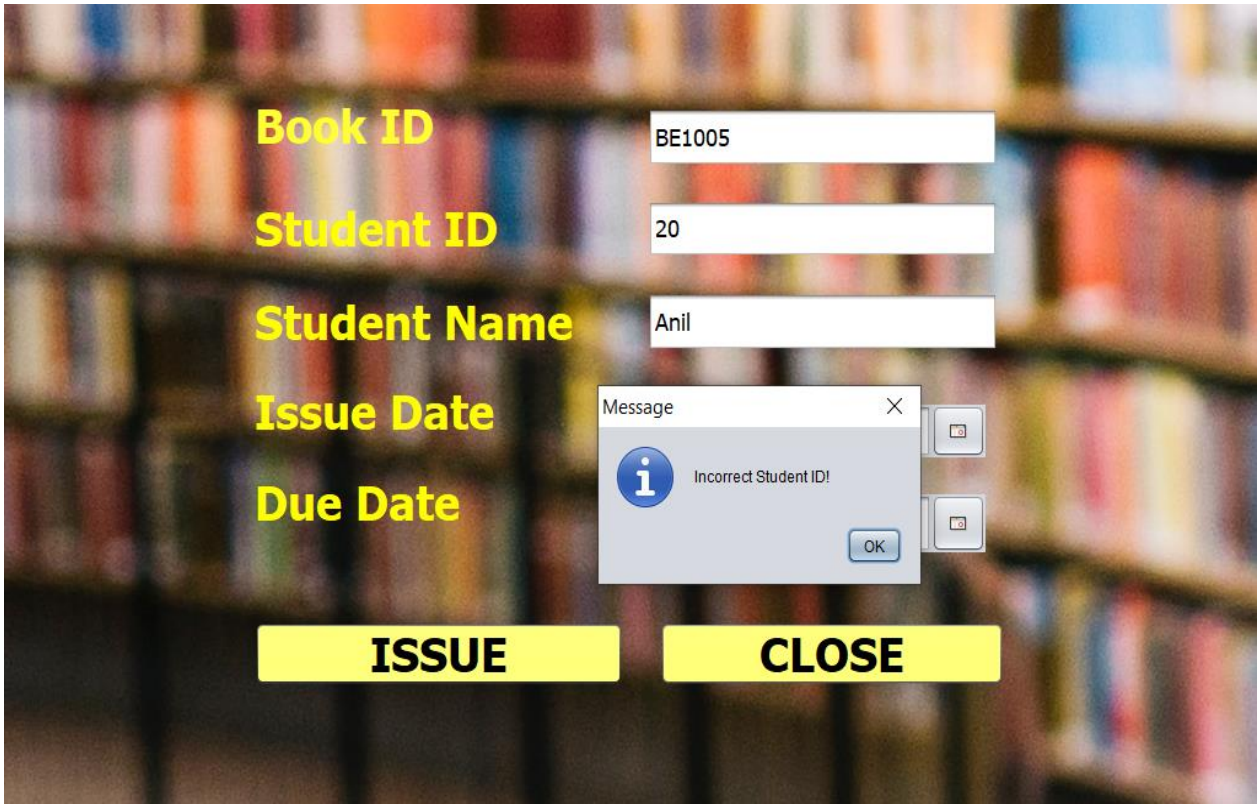

Book ID
CA1001
Book Name
GATE leaf
Publisher
Arvind Kovil
Book Price
1250
Publishing Year
Message
Book ID already exists!
OK
SAVE
CLOSE

- ***Screenshot for Issue Book Form***

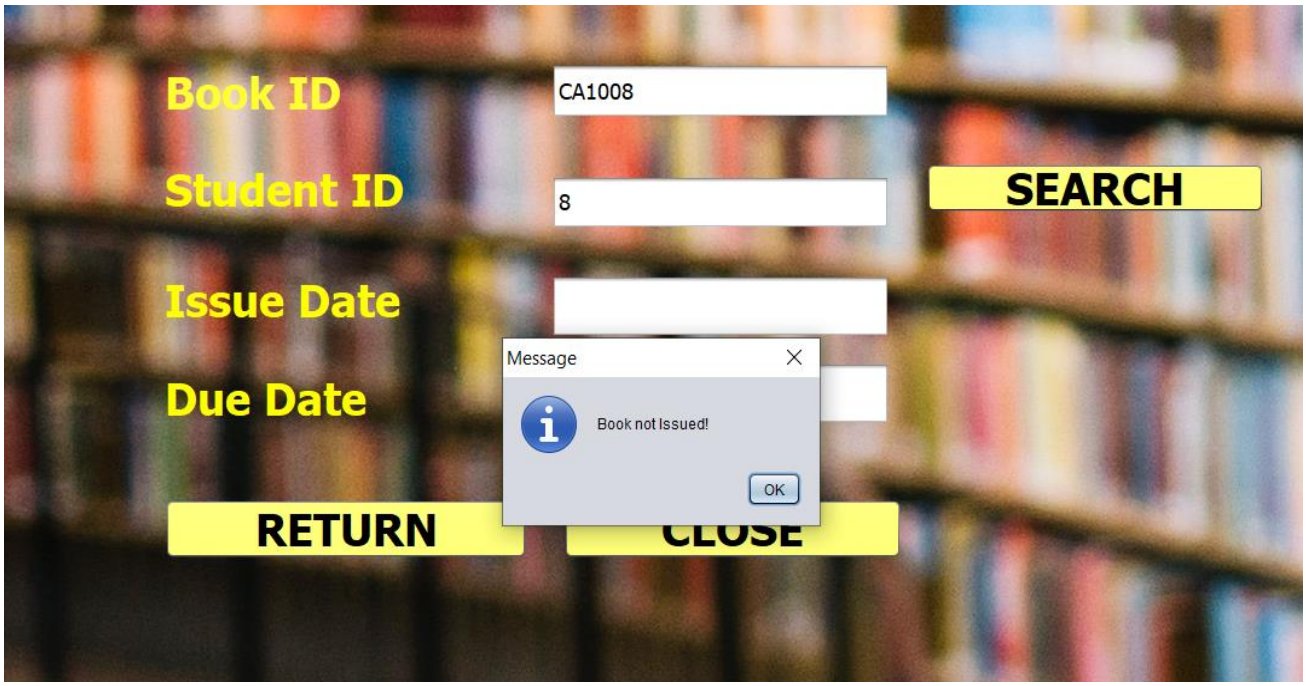


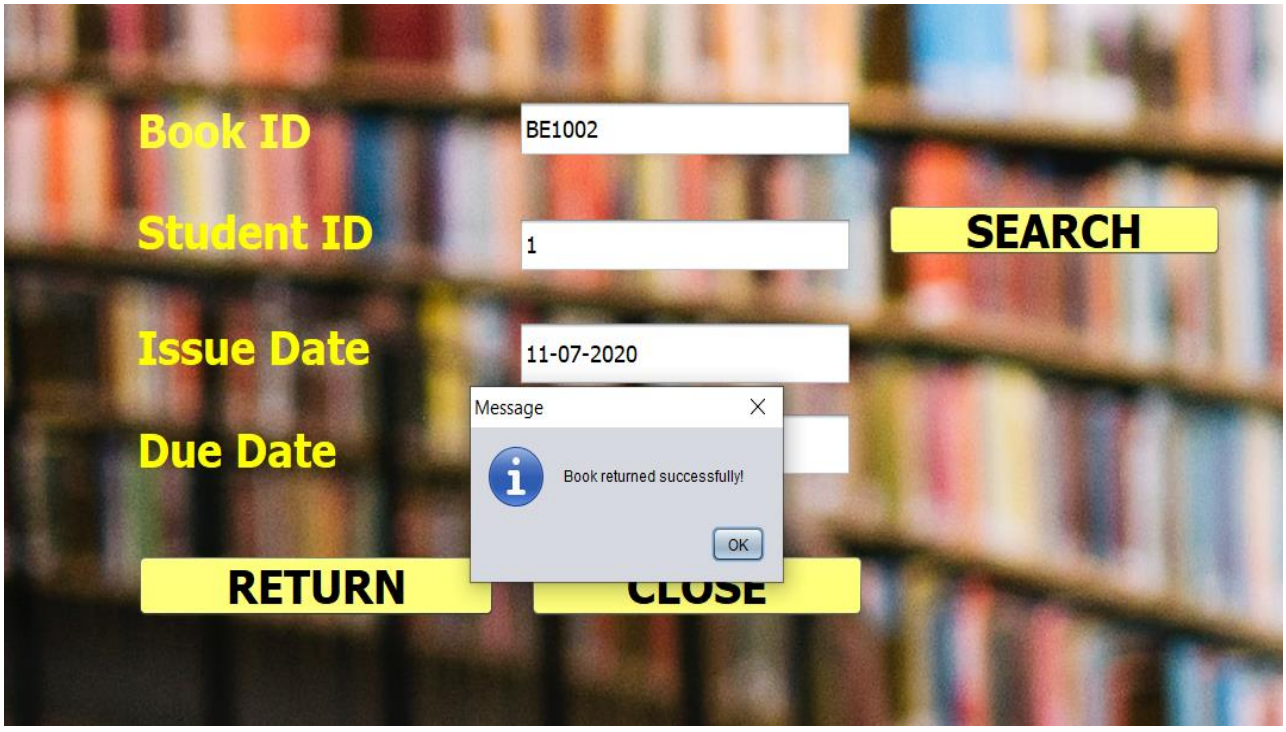



Book ID
CA1005
Student ID
18
Student Name
Reema Shukla
Issue Date
Due Date
Message
Book issued successfully!
OK
ISSUE
CLOSE

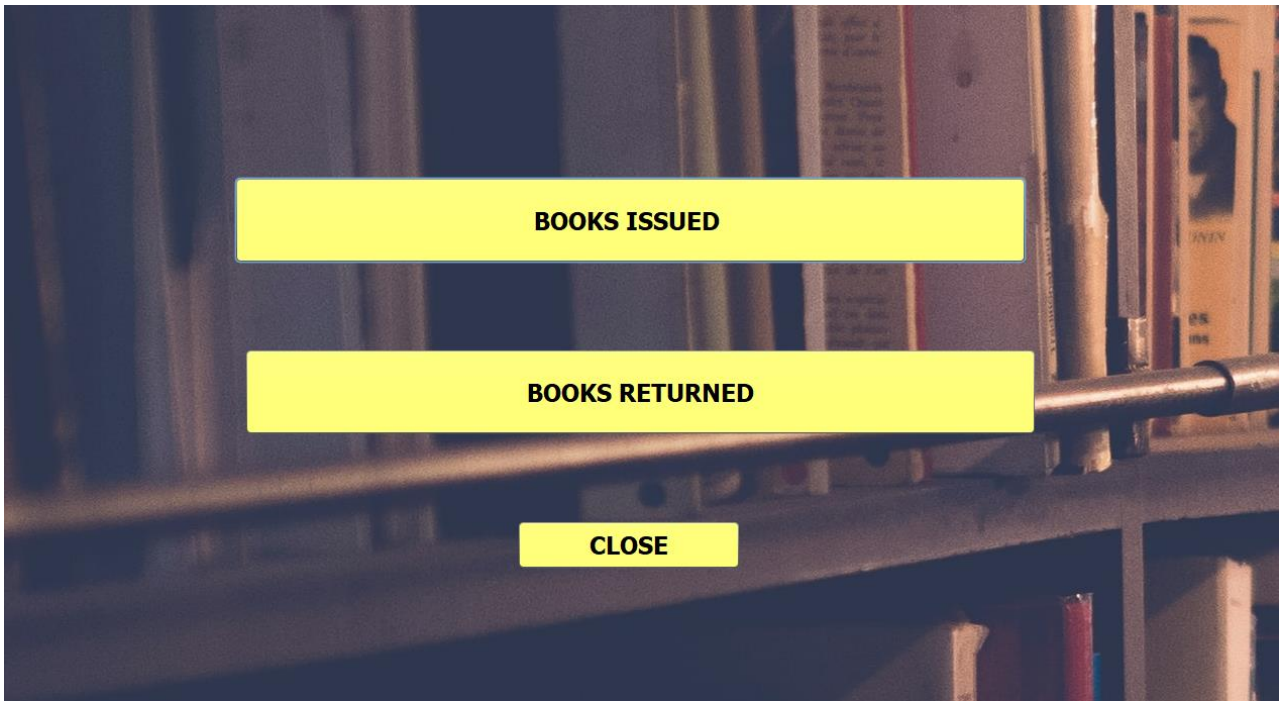

Book ID
BE1005
Student ID
20
Student Name
Anil
Issue Date
Due Date
Message
Incorrect Student ID!
OK
ISSUE
CLOSE

- ***Screenshot for Return Book Form***

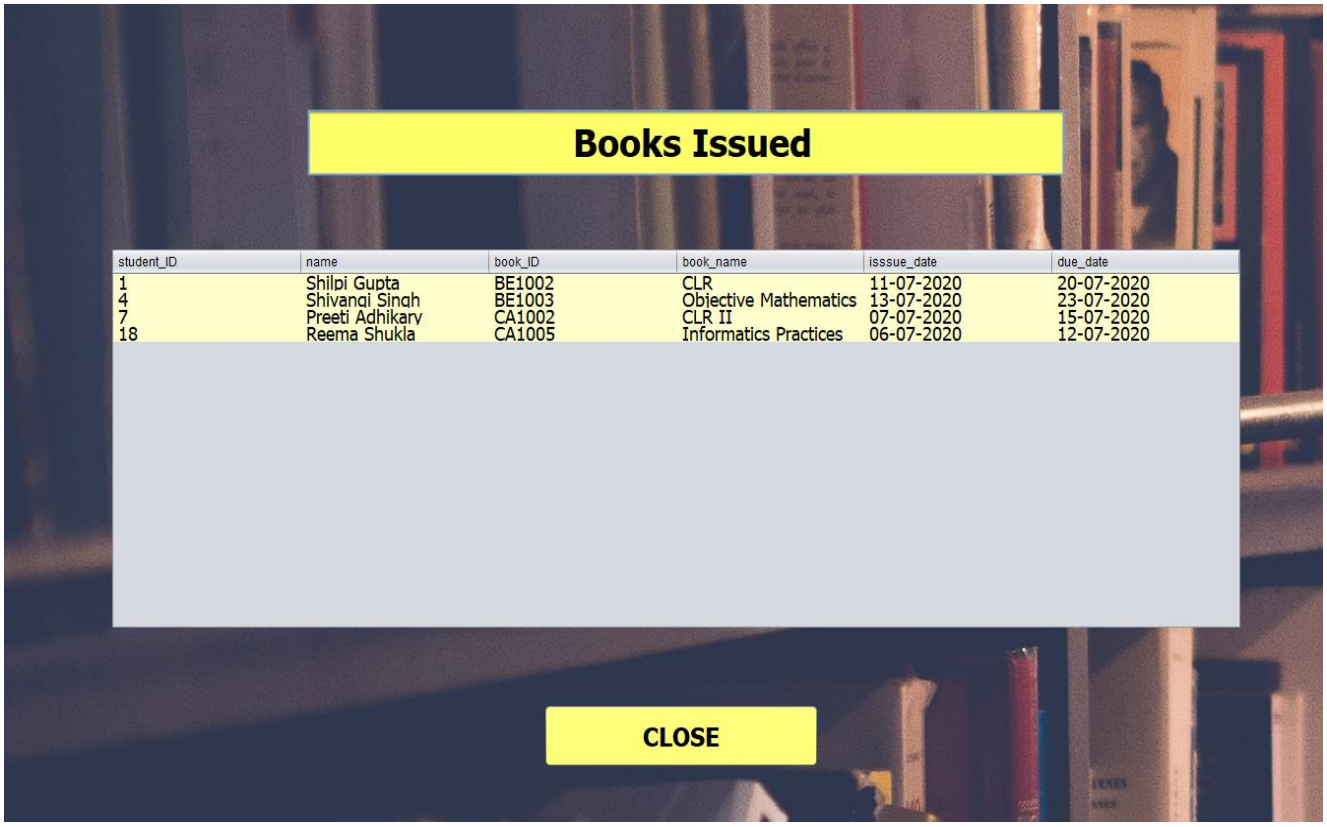




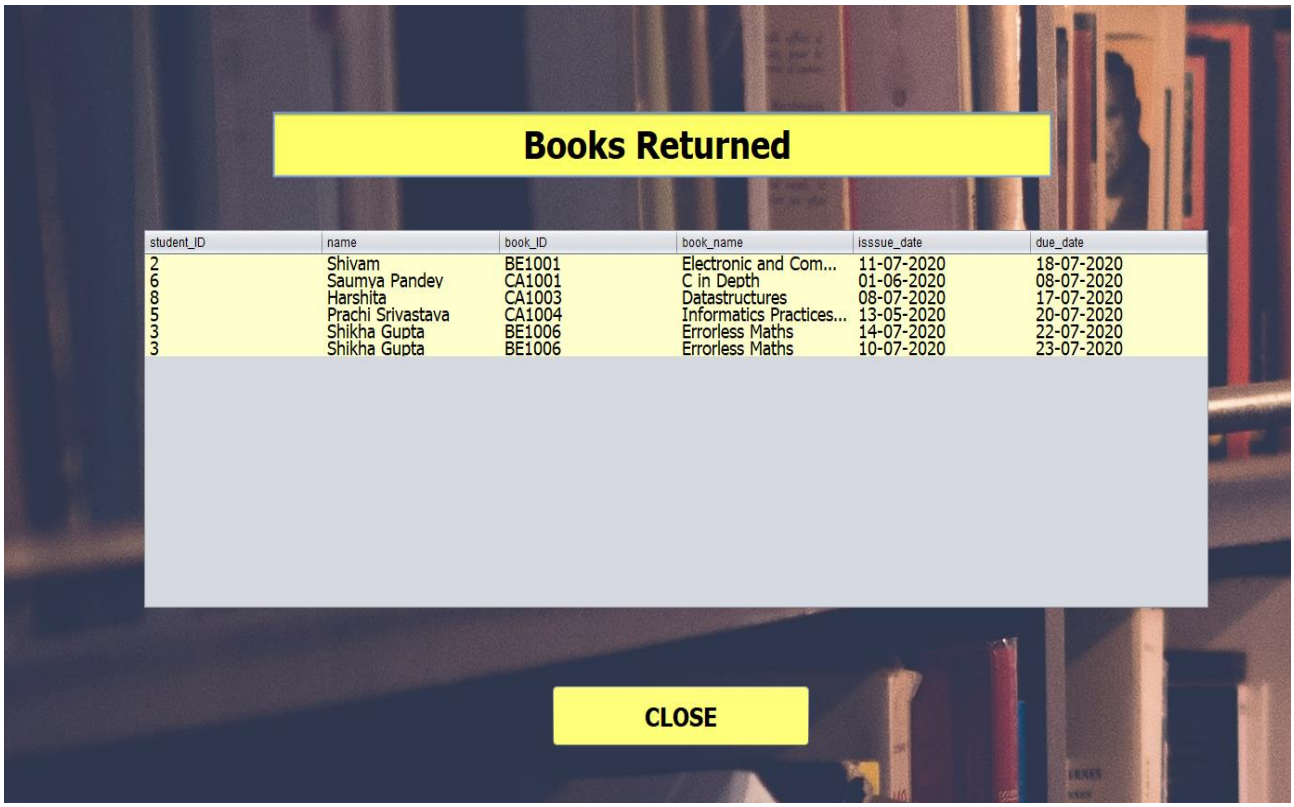
- ***Screenshot for Details Form***



- ***Screenshot for Issue Details Form***



- ***Screenshot for Return Details Form***



SQL TABLE CONTENTS AND DESCRIPTION

Table: student

- ***Desc student;***

```
mysql> desc student;
+------------+--------------+------+-----+---------+-------+
| Field      | Type         | Null | Key | Default | Extra |
+------------+--------------+------+-----+---------+-------+
| STUDENTID  | varchar(10)  | NO   | PRI | NULL    |       |
| NAME       | varchar(100) | YES  |     | NULL    |       |
| FATHERNAME | varchar(100) | YES  |     | NULL    |       |
| COURSENAME | varchar(30)  | YES  |     | NULL    |       |
| BRANCHNAME | varchar(30)  | YES  |     | NULL    |       |
+------------+--------------+------+-----+---------+-------+
5 rows in set (0.17 sec)
```

- ***Select * from student;***

```
mysql> select * from student;
+-----------+------------------+----------------------+------------+----------------------+
| STUDENTID | NAME             | FATHERNAME           | COURSENAME | BRANCHNAME           |
+-----------+------------------+----------------------+------------+----------------------+
| 1         | Shilpi Gupta     | Ashwini Kumar        | MCA        | Computer Applications |
| 10        | Mahima Pandey    | M. Pandey            | MCA        | Computer Applications |
| 11        | Avinash Gupta    | A.K. Gupta           | BTech      | CSE                  |
| 12        | Vipin Pal        | B.P. Singh           | Ph.D       | Prod                 |
| 13        | Sachin Kumar Pal | Bhanu Pratap         | BTech      | ME                   |
| 14        | Anamika Srivastav | Ashok Srivastav      | BTech      | Electrical           |
| 15        | Simran Gupta     | G. Gupta             | B.Arch     | Prod                 |
| 16        | Abhisek Gupta    | Ishwar Chanda        | BTech      | ME                   |
| 17        | Shreya Gupta     | Ishwar Gupta         | M.Arch     | Prod                 |
| 18        | Reema Shukla     | Ankit Shukla         | B.Arch     | ME                   |
| 2         | Shivam           | Chetan Pal           | MCA        | Computer Applications |
| 3         | Shikha Gupta     | Ashwini Kumar        | Ph.D       | CSE                  |
| 4         | Shivangi Singh   | Anil Singh           | BTech      | EEC                  |
| 5         | Prachi Srivastava | Ajay Kumar Srivastava | BTech      | Electrical           |
| 6         | Saumya Pandey    | Rajesh Pandey        | MTech      | CSE                  |
| 7         | Preeti Adhikary  | Gautam Adhikary      | BTech      | IT                   |
| 8         | Harshita         | Jha                  | MCA        | Computer Applications |
| 9         | Ankita Gupta     | M. Gupta             | MCA        | Computer Applications |
+-----------+------------------+----------------------+------------+----------------------+
18 rows in set (0.00 sec)
```

Table: addbook

- ***Desc addbook;***

```
mysql> desc addbook;
+-----------------+--------------+------+-----+---------+-------+
| Field           | Type         | Null | Key | Default | Extra |
+-----------------+--------------+------+-----+---------+-------+
| book_id         | varchar(10)  | NO   | PRI | NULL    |       |
| BOOK_NAME       | varchar(100) | YES  |     | NULL    |       |
| PUBLISHER       | varchar(100) | YES  |     | NULL    |       |
| BOOK_PRICE      | varchar(10)  | YES  |     | NULL    |       |
| PUBLISHING_YEAR | varchar(100) | YES  |     | NULL    |       |
+-----------------+--------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

- ***Select * from addbook;***

```
mysql> select * from addbook;
+---------+-----------------------------+-------------------+------------+-----------------+
| book_id | BOOK_NAME                   | PUBLISHER         | BOOK_PRICE | PUBLISHING_YEAR |
+---------+-----------------------------+-------------------+------------+-----------------+
| BE1001  | Electronic and Communication | Penguin           | 1050       | 2017            |
| BE1002  | CLR                         | Penguin           | 2000       | 2001            |
| BE1003  | Objective Mathematics       | S. Chand          | 1099       | 2011            |
| BE1004  | Computer fundamentals in C  | Reema Thareja     | 495        | 2005            |
| BE1005  | Fundamentals of ICs         | S. Chand          | 978        | 2011            |
| BE1006  | Errorless Maths             | K. N. Murthy      | 786        | 2001            |
| BE1007  | Informatica                 | S.K. Srivastava   | 993        | 1995            |
| CA1001  | C in Depth                  | Deepali Srivastava | 878        | 2009            |
| CA1002  | CLR II                      | A.K. John         | 2050       | 2005            |
| CA1003  | Datastructures              | Anne K. Alan      | 695        | 1998            |
| CA1004  | Informatics Practices(Java) | Sumita Arora      | 991        | 2011            |
| CA1005  | Informatics Practices       | Sumita Arora      | 991        | 2012            |
| CA1006  | DAA                         | K. N. Shah        | 1001       | 1995            |
| CA2451  | CAT Guide                   | Arun Sharma       | 786        | 1991            |
+---------+-----------------------------+-------------------+------------+-----------------+
14 rows in set (0.00 sec)
```

Table: issue

- ***Desc issue;***

```
mysql> desc issue;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| BOOK_ID     | varchar(10) | YES  |     | NULL    |       |
| STUDENT_ID  | varchar(10) | YES  |     | NULL    |       |
| NAME        | varchar(30) | YES  |     | NULL    |       |
| ISSSUE_DATE | varchar(20) | YES  |     | NULL    |       |
| DUE_DATE    | varchar(20) | YES  |     | NULL    |       |
| RETURN_BOOK | varchar(5)  | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

- ***Select * from issue;***

```
mysql> select * from issue;
+---------+------------+------------------+-------------+------------+-------------+
| BOOK_ID | STUDENT_ID | NAME             | ISSSUE_DATE | DUE_DATE   | RETURN_BOOK |
+---------+------------+------------------+-------------+------------+-------------+
| BE1001  | 2          | Shivam           | 11-07-2020  | 18-07-2020 | Yes         |
| BE1002  | 1          | Shilpi Gupta     | 11-07-2020  | 20-07-2020 | No          |
| BE1003  | 4          | Shivangi Singh   | 13-07-2020  | 23-07-2020 | No          |
| CA1001  | 6          | Saumya Pandey    | 01-06-2020  | 08-07-2020 | Yes         |
| CA1002  | 7          | Preeti Adhikary  | 07-07-2020  | 15-07-2020 | No          |
| CA1003  | 8          | Harshita         | 08-07-2020  | 17-07-2020 | Yes         |
| CA1004  | 5          | Prachi Srivastava| 13-05-2020  | 20-07-2020 | Yes         |
| BE1006  | 3          | Shikha Gupta     | 14-07-2020  | 22-07-2020 | Yes         |
| BE1006  | 3          | Shikha           | 10-07-2020  | 23-07-2020 | Yes         |
| CA1005  | 18         | Reema Shukla     | 06-07-2020  | 12-07-2020 | No          |
+---------+------------+------------------+-------------+------------+-------------+
10 rows in set (0.00 sec)
```

CODING

➢ Login1.java

- ***On Close button:***

```
System.exit(0);
```

- ***On Login button:***

```
if(jTextField1.getText().equals("root") &&
jPasswordField1.getText().equals("root123"))
        {
           setVisible(false);
           new home().setVisible(true);
        }
        else
           JOptionPane.showMessageDialog(null,"Username or
Password is Invalid! Try Again.");
```

➢ home.java

- ***On Add Student button:***

```
new addstudent().setVisible(true);
```

- ***On Add Book button:***

```
new addbook().setVisible(true);
```

- ***On Issue Book button:***

new issuebook().setVisible(true);

- ***On Return Book button:***

  new returnbook().setVisible(true);

- ***On Details button:***

  new details1().setVisible(true);

- ***On Logout button:***

  new login1().setVisible(true);

➢ addstudent.java

- ***On Close button:***

  SetVisible(false);

- ***On Save button:***

```
String studentID=jTextField1.getText();
        String name=jTextField2.getText();
        String fatherName=jTextField3.getText();
        String courseName=(String)jComboBox1.getSelectedItem();
        String branchName=(String)jComboBox2.getSelectedItem();

        try
        {
           Connection con=ConnectionProvider.getCon();
           Statement st=con.createStatement();
```

```
            st.executeUpdate("insert into student
values('"+studentID+"','"+name+"','"+fatherName+"','"+courseNam
e+"','"+branchName+"')");
            JOptionPane.showMessageDialog(null,"Student details
successfully added!");
            setVisible(false);
            new addstudent().setVisible(true);

        }
        catch(Exception e)
              {
                 JOptionPane.showMessageDialog(null,"Student ID
already exists!");
                  setVisible(false);
                 new addstudent().setVisible(true);

              }
```

➢ addbook.java

- ***<u>On Close button:</u>***

  SetVisible(false);

- ***<u>On Save button:</u>***

```
String bookID=jTextField1.getText();
      String name=jTextField2.getText();
      String publisher=jTextField3.getText();
      String price=jTextField4.getText();
      String publishingYear=jTextField5.getText();
      try
      {
         Connection con=ConnectionProvider.getCon();
         Statement st=con.createStatement();
```

```
            st.executeUpdate("insert into addbook
values('"+bookID+"','"+name+"','"+publisher+"','"+price+"','"+publ
ishingYear+"')");
            JOptionPane.showMessageDialog(null,"Book details
successfully updated!");
            setVisible(false);
            new addbook().setVisible(true);
        }
        catch(Exception e)
              {

                  JOptionPane.showMessageDialog(null,"Book ID
already exists!");
                   setVisible(false);
                  new addbook().setVisible(true);
              }
```

➢ issuebook.java

- ***On Close button:***

SetVisible(false);

- ***On Issue button:***

```
  SimpleDateFormat dFormat=new SimpleDateFormat("dd-MM-
yyyy");
        String bookID=jTextField1.getText();
        String studentID=jTextField2.getText();
        String name=jTextField3.getText();
        String issueDate=dFormat.format(jDateChooser1.getDate());
        String dueDate=dFormat.format(jDateChooser2.getDate());
        String returnBook="No";

        try
        {
           Connection con=ConnectionProvider.getCon();
```

```
            Statement st=con.createStatement();
            ResultSet rs=st.executeQuery("select * from addbook
where book_ID='"+bookID+"'");
            if(rs.next())
            {
               ResultSet rsl=st.executeQuery("select * from student
where studentID='"+studentID+"'");
               if(rsl.next())
               {
                st.executeUpdate("insert into issue
values('"+bookID+"','"+studentID+"','"+name+"','"+issueDate+"','"
+dueDate+"','"+returnBook+"')");
                 JOptionPane.showMessageDialog(null,"Book issued
successfully!");
                  setVisible(false);
                  new issuebook().setVisible(true);
               }
               else
                  JOptionPane.showMessageDialog(null,"Incorrect
Student ID!");
            }
            else
            JOptionPane.showMessageDialog(null,"Incorrect Book
ID!");


         }
         catch(Exception e)
         {

            JOptionPane.showMessageDialog(null,"Connection
Error!");

         }
```

➢ returnbook.java

- ***<u>On Close button:</u>***

```
setVisible(false);
```

- ***<u>On Return button:</u>***

```
String bookID=jTextField1.getText();
        String studentID=jTextField2.getText();

        try
        {
            Connection con=ConnectionProvider.getCon();
            Statement st=con.createStatement();
            st.executeUpdate("update issue set return_book='Yes'
where book_ID='"+bookID+"' and student_ID='"+studentID+"'");
            JOptionPane.showMessageDialog(null,"Book returned
successfully!");
          setVisible(false);
          new returnbook().setVisible(true);

        }
        catch(Exception e)
        {

            JOptionPane.showMessageDialog(null,"Connection
Error!");

        }
```

- ***<u>On Search button:</u>***

```
String bookID=jTextField1.getText();
    String studentID=jTextField2.getText();
    try {
        Connection con = ConnectionProvider.getCon();
        Statement st = con.createStatement();
         ResultSet rs=st.executeQuery("select * from issue where
book_ID='"+bookID+"'and student_ID='"+studentID+"'");
     if(rs.next())
        {
           jTextField4.setText(rs.getString(4));
           jTextField5.setText(rs.getString(5));
           jTextField1.setEditable(false);
           jTextField2.setEditable(false);

        }

     else
     {  JOptionPane.showMessageDialog(null,"Book not
Issued!");
         setVisible(false);
         new returnbook().setVisible(true);

      }
    }
     catch(Exception e)
    {

       JOptionPane.showMessageDialog(null,"Connection
Error!");

    }
```

➢ details1.java

- ***On Books Issued button:***

  new idetails().setVisible(true);

- ***On Books Returned button:***

  new rdetails().setVisible(true);

- ***On Close button:***

  setVisible(false);

➢ idetails.java

- ***On Close button:***

  setVisible(false);

- ***formComponentShown:***

```
try
    {
       Connection con=ConnectionProvider.getCon();
       Statement st=con.createStatement();
       ResultSet rs=st.executeQuery("select
issue.student_ID,student.name,issue.book_ID,addbook.book_nam
e,issue.isssue_date,issue.due_date from student inner join addbook
inner join issue where addbook.book_ID=issue.book_ID and
student.studentID=issue.student_ID and issue.return_book='No'");
       jTable1.setModel(DbUtils.resultSetToTableModel(rs));
    }
```

```
        catch(Exception e)
        {
          JOptionPane.showMessageDialog(null,"Connection
Error");
        }
```

➢ rdetails.java

- ***On Close button:***

```
setVisible(false);
```

- ***formComponentShown:***

```
 try
        {
           Connection con=ConnectionProvider.getCon();
           Statement st=con.createStatement();
           ResultSet rs=st.executeQuery("select
issue.student_ID,student.name,issue.book_ID,addbook.book_nam
e,issue.isssue_date,issue.due_date from student inner join addbook
inner join issue where addbook.book_ID=issue.book_ID and
student.studentID=issue.student_ID and
issue.return_book='Yes'");
           jTable1.setModel(DbUtils.resultSetToTableModel(rs));
        }
        catch(Exception e)
        {
          JOptionPane.showMessageDialog(null,"Connection
Error");
        }
```

CONCLUSION & FUTURE SCOPE

This application provides a computerized version of library management system which will benefit the students as well as the staff of the library.

It makes entire process computerized where staff can generate reports and do book transactions. It has a facility of Librarian login where he/she can add newly admitted students details in a digitalized form. He can also add new books. He can issue books to the students at one click. Book returning procedure also becomes very easy by this application.

Librarian has the authority to see the details of the books issued and books returned in a span of time. This application reduces paper work to a good extend. It helps in saving paper and hence contributes in to a healthy environment.

There is a future scope of this facility that many more features such as online lectures video tutorials can be added by teachers as well as online assignments submission facility , a feature of group chat where students can discuss various issues of engineering can be added to this project thus making it more interactive more user friendly and project which fulfills each users need in the best way possible.

REFERENCES

- *INFORMATICS PRACTICES*
  -by Sumita Arora
- *www.google.com*
- *http://www.w3schools.com/sql/sql_insert.asp*
- *http://www.w3schools.com/sql/sql_update.asp*

➢ ***WEBSITES USED FOR PICTURES:-***

- *www.google.com/images*
- *www.yahoo.com*