Project Report on

LIBRARY MANAGEMENT SYSTEM

Web Technology Lab-3CS3403

National Institute of Technology, Patna

Submitted to: Dr. Amit Kumar Singh

Submitted by: M. Sampath Kumar (1806142)

P. Krishna Kumar (1806114)

Table of Content

1.Objective

2. Introduction

3. Users of the system

4. User Flow Chart

5. Admin Flow Chart

6. References

Objective:

- This system can be used to search for books or magazines, reserve books, put request to buy new book.

- This is one integrated system that contains both user component and the librarian component.

- User can get the latest updates which are sent by the librarian.

Introduction:

This project "*Library Management System*" is developed in HTML, CSS, JS, Servlets, JAVA. This project is for monitoring and controlling the library transactions. This includes basic operations like search for the required book, adding a new member, adding new book, updating the latest information in the library, and has the facility to borrow and return books from the library.

There are two types of users for the website.

1. Student
2. Admin

- The student can be registered by his/her own i.e, he/she can create his/her own account and can login safely and access the data available.
- Where as the admin is already registered and only a few admins are registered and they have the full authority to access the full data of the library.
- Student can request a new book that is required for him/her and can pre-reserve a book and some other functionalities.

Users of the System:

- Student:

  Every student can create his/her own account in the website and can see his/her details from the library side. He/she can access the facilities provided by the admin.

  He/she can see the full data about him/her like the book details that he borrowed and the details of the books about return date, fine (if any late return).

  He/she can request the book that is not available in the library as per the requirement and can pre-reserve a book and see if the book required to him/her is present int e library or not.

- Admin:

  Admin (librarian) can see the whole library data and manage the whole data that a user (Student) can see when he/she will login to their respective accounts.

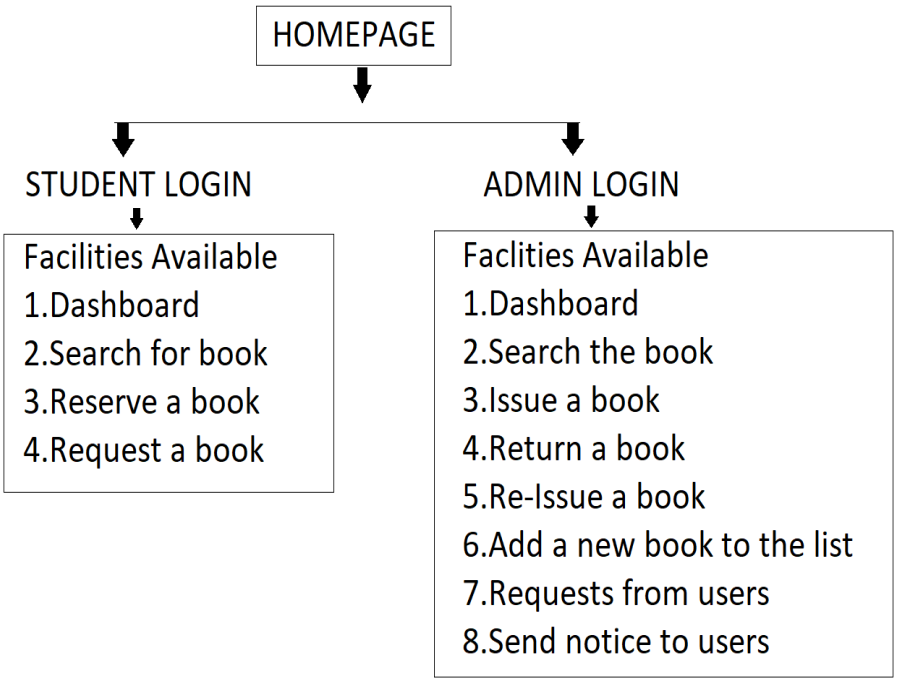
  Admin can see the requests that are sent by the students and can send the notice about the

new book if any, and issue and borrow the books from the students.

He has the full data of all the books that are in the library the details of the student with whom a particular book is present.

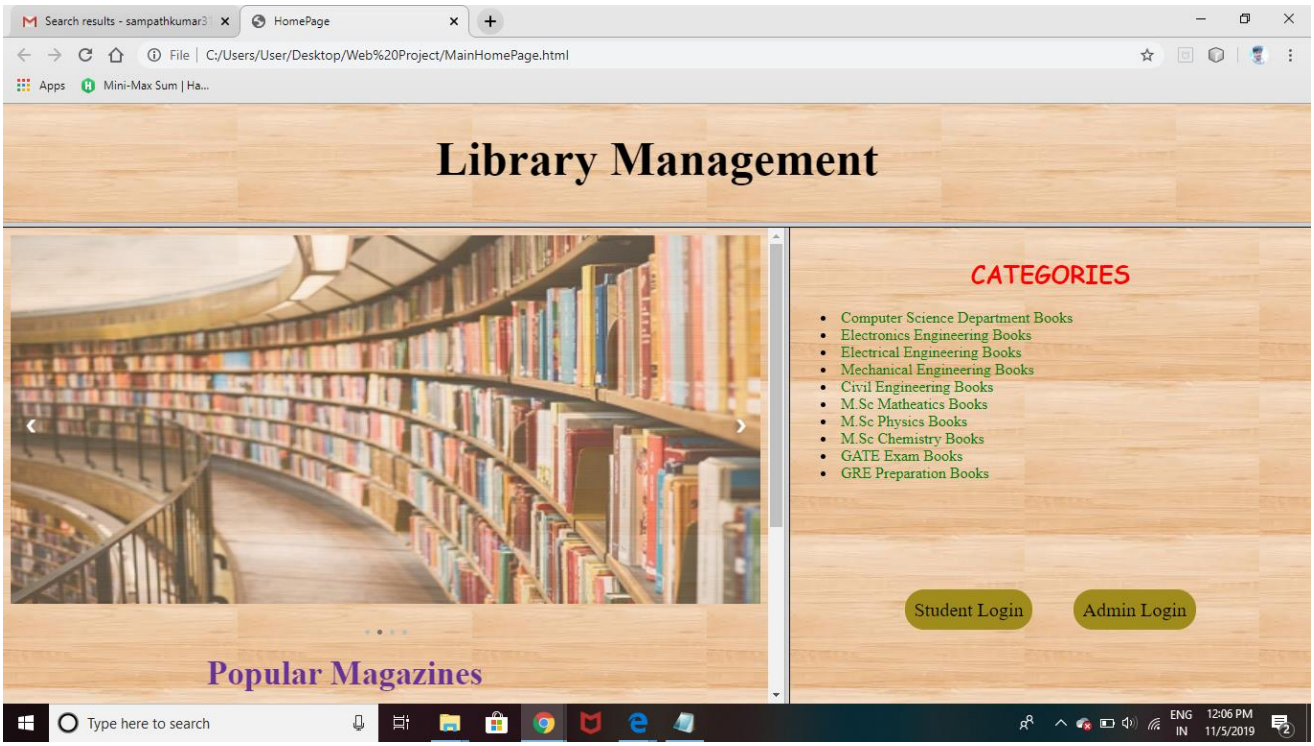
He can access the whole data base and can edit the details of the student or the particular book.

Main flow chart:

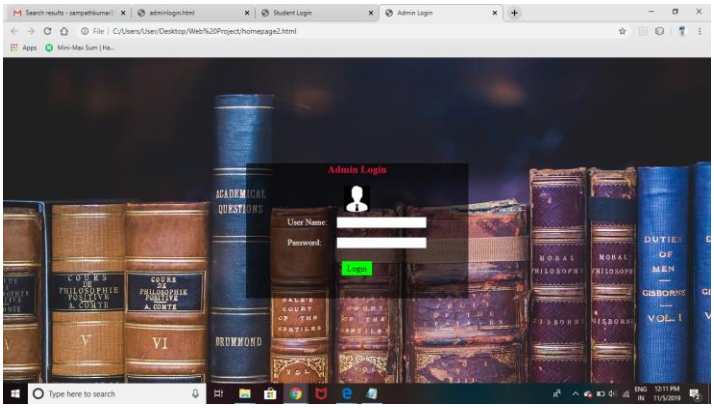


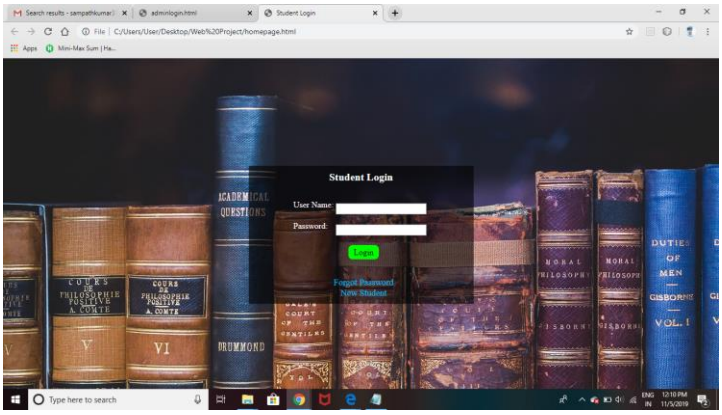
Screenshots:

Homepage:

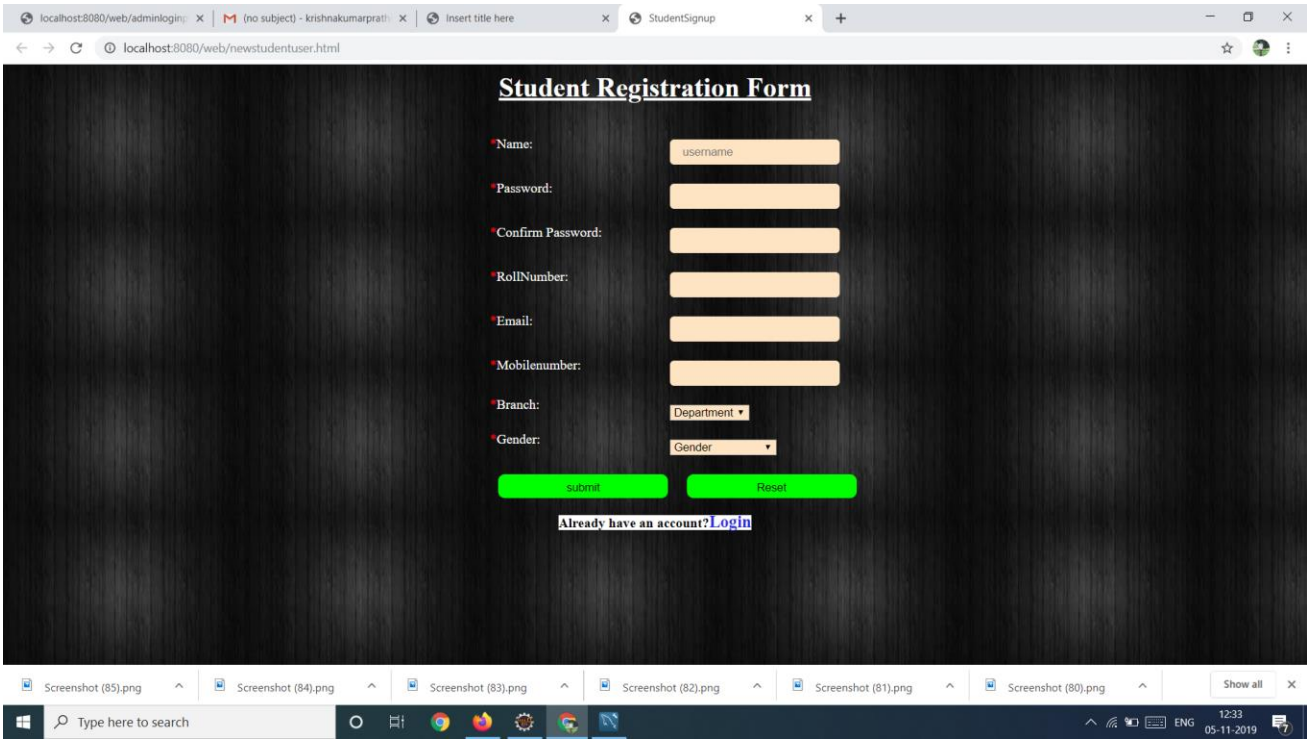


User and Admin Login pages:



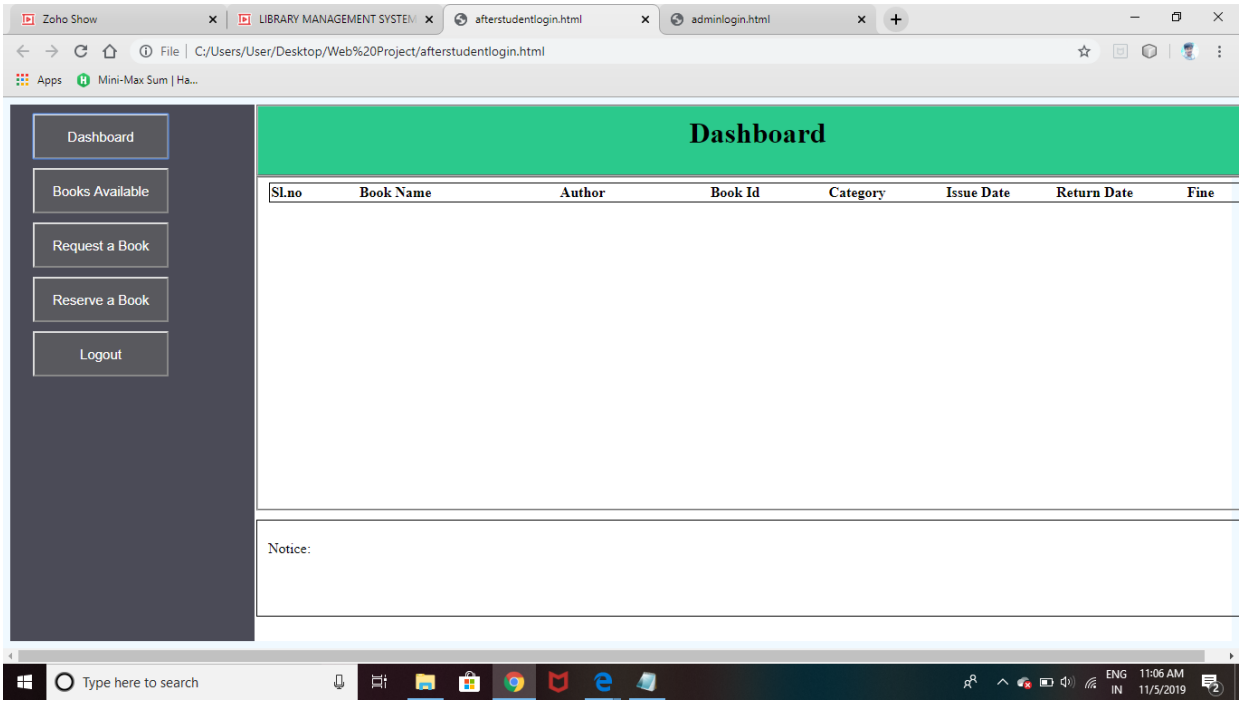


Student Registration form:

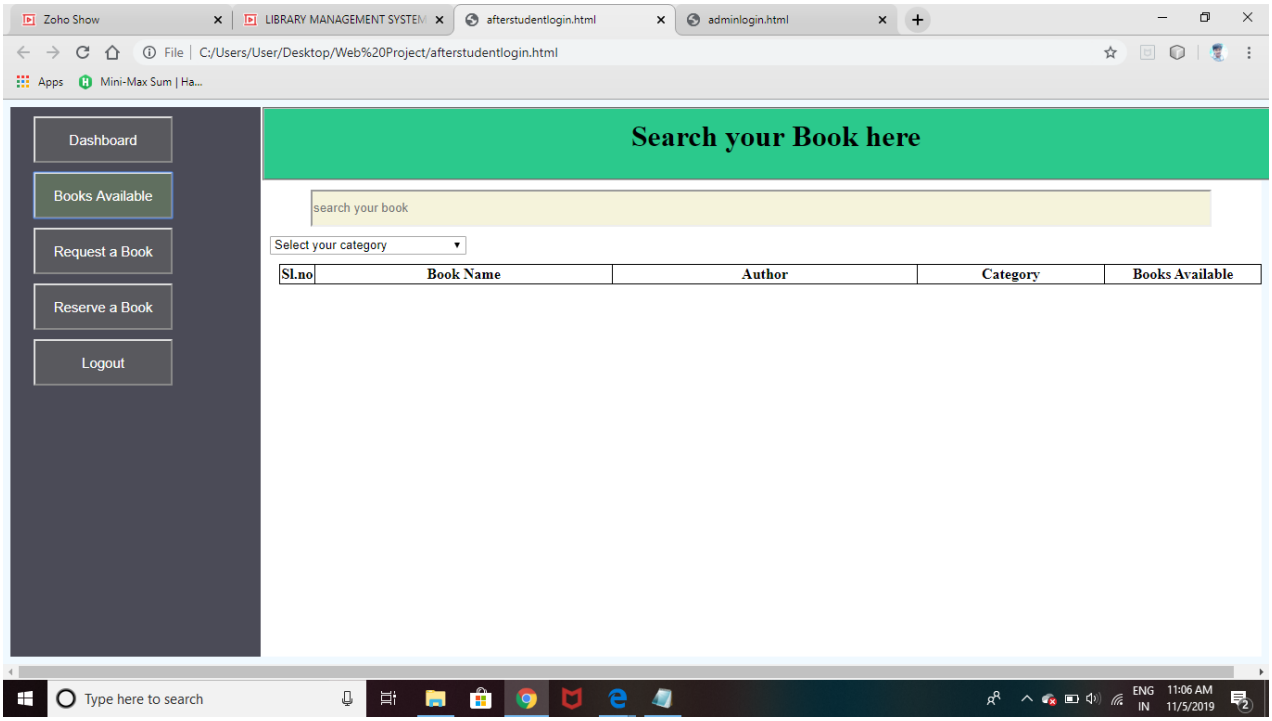


User Flow chart:

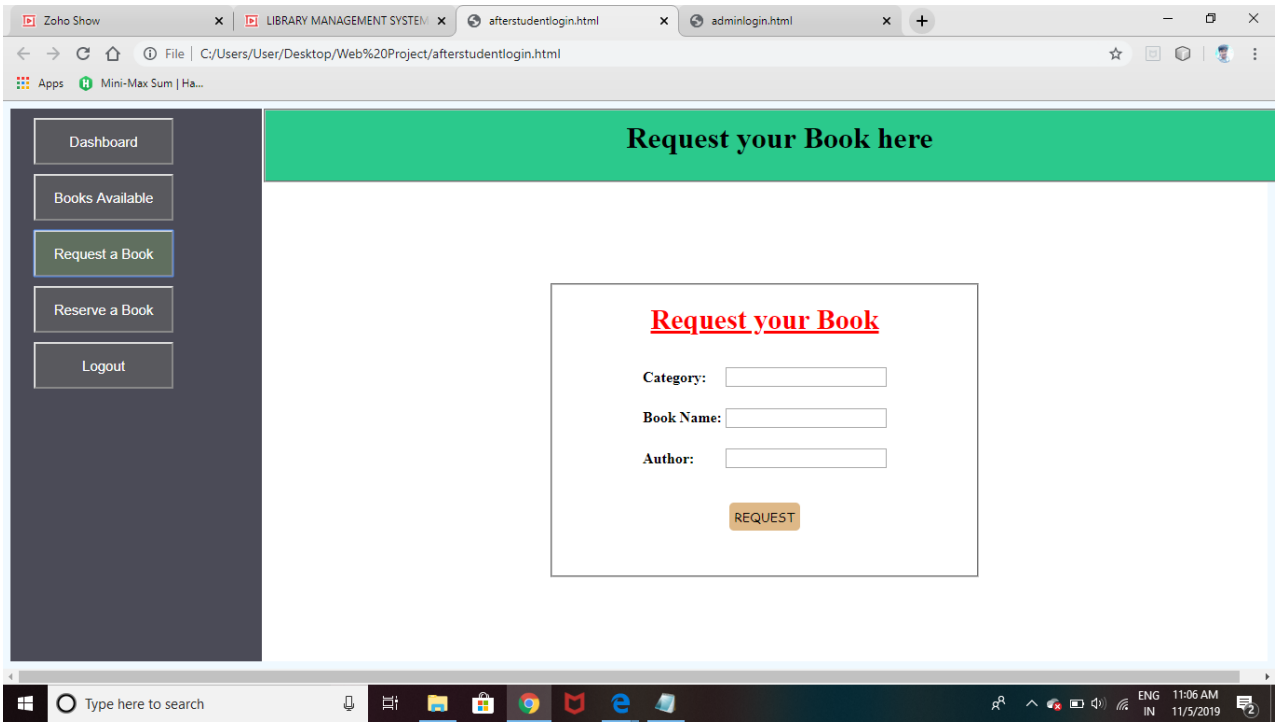
User Dashboard:



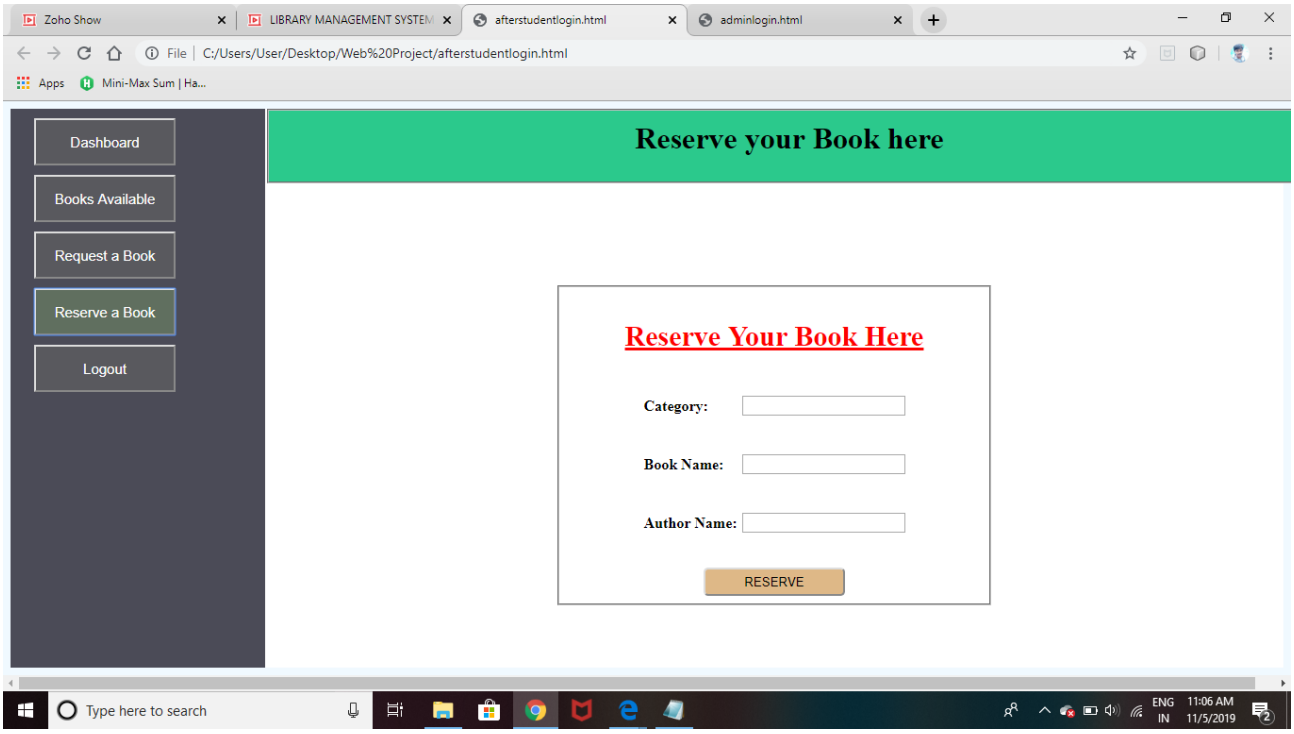
Books Available page:



Request a book page:

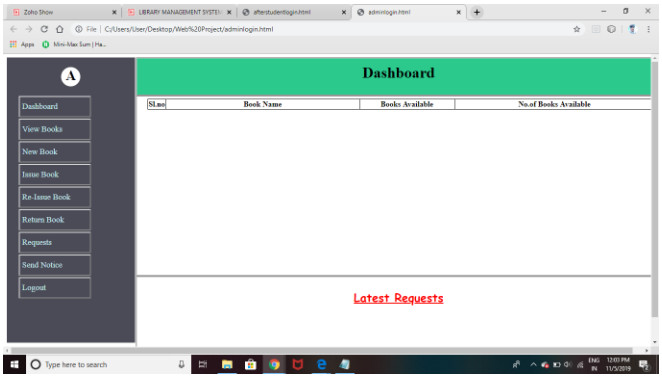


Reserve book page:

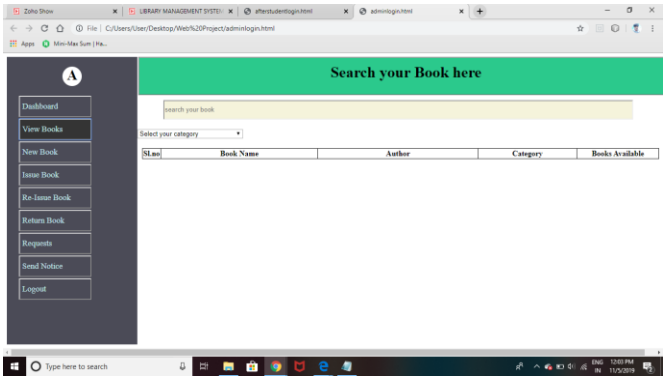


Admin Flow Chart:

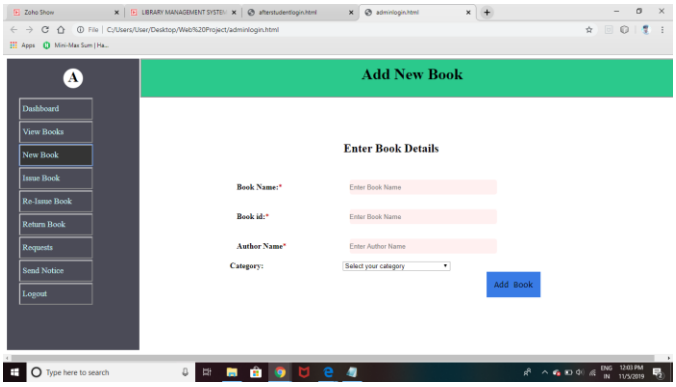
Admin Dashboard:



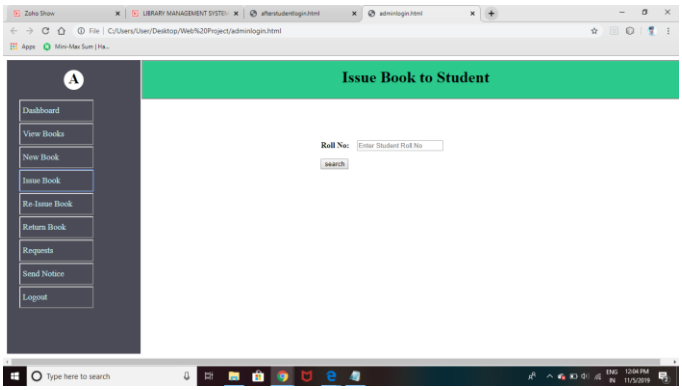
Admin Search books page:



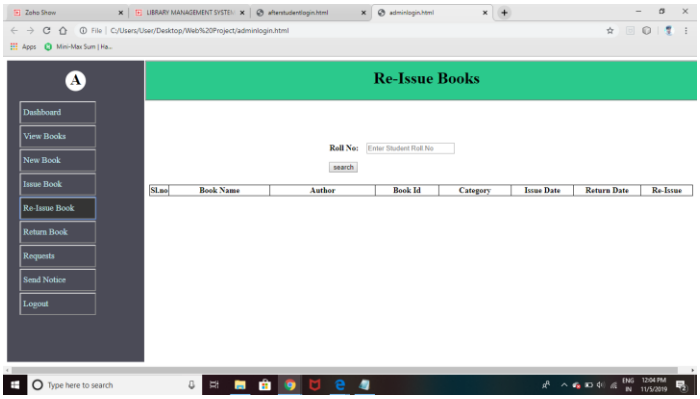
Add new book page:



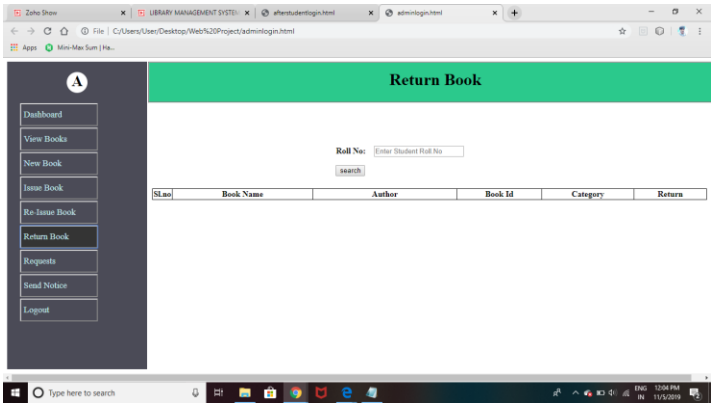
Issue Book page:



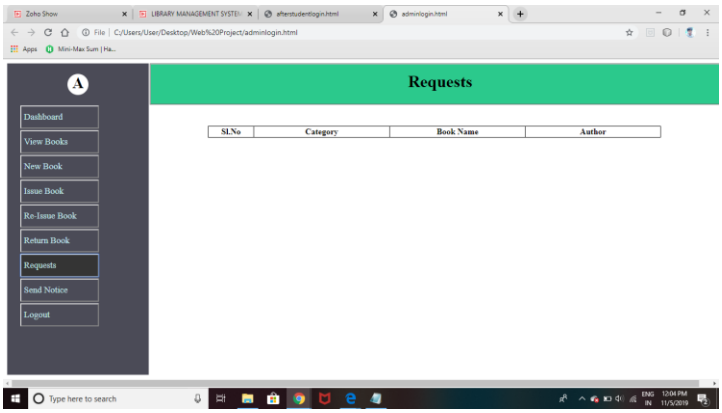
Re-Issue Book page:



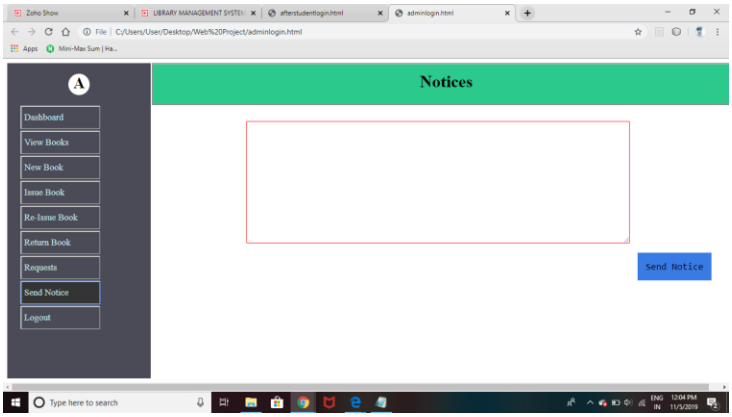
Return book page:



Requests from the users page:



Notices page:



Programme: (example)

```
package com.web;

 import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.time.LocalDateTime;

import java.time.LocalTime;

 import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;


@WebServlet("/Studentdata")

public class Studentdata extends HttpServlet {

   String url="jdbc:mysql://localhost:3306/webp";

   String username="krishna";

   String password="Vamsi@123";

   protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

       try{

      String uname=request.getParameter("username");

       String pass=request.getParameter("pas");

       String email=request.getParameter("ema");
```

```
        String mobilenumber=request.getParameter("mobilenumber");
        String rollnumber=request.getParameter("number");
        String branch=request.getParameter("branch");
        String gender=request.getParameter("gender");
        LocalDateTime issuedate=(java.time.LocalDateTime.now());
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection(url,username,password);
         String query = "insert into login(username,password,email,gender,branch,rollnumber)
values(?,?,?,?,?,?)";
           PreparedStatement ps = con.prepareStatement(query);
           ps.setString(1, uname);
            ps.setString(2, pass);
            ps.setString(3, email);
            ps.setString(4, gender);
            ps.setString(5, branch);
            ps.setString(6, rollnumber);
            ps.executeUpdate();
        PrintWriter out=response.getWriter();
        out.println("Successfully Registered");
         ps.close();
            con.close();
        }catch(Exception e)
        {System.out.println(e);}
    }
}
```

References:

- W3Schools.com
- Stackoverflow.com
- Tutorialspoint.com
- Head first Servlets and JSP
- Web Technologies-Uttam K. Roy ,Oxford Publications

---THE END---