CS353 Database Systems - Spring 2022

Group 24

School Library System

Project URL: https://ozgur-abi.github.io/CS353-SchoolLibraryDatabase/

Ali Doğaç Urkaya - 21903213 - Section 3

Özgür Abi - 21902358 - Section 3

Jankat Berslan Dinçer - 21902035 - Section 3

Serhat Gürgenyatağı - 21903288 - Section 3

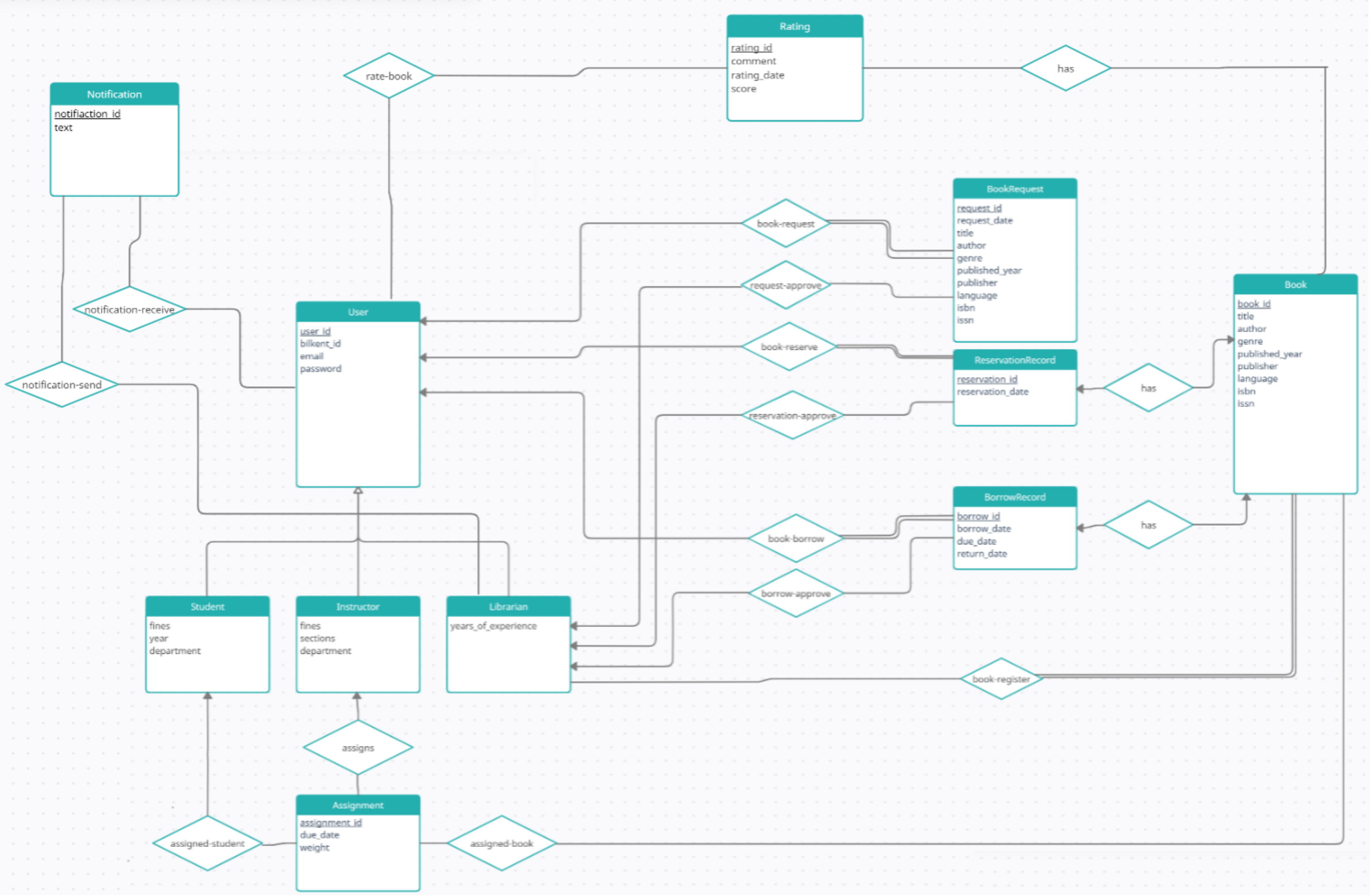
Table of Contents

1. Description

In this project, we have designed a library system for the university. This system can be used by students, instructors, and librarians. Its aim is to provide an application where these 3 groups can communicate on a recorded basis regarding the library books. Here are the some of the key features:

- Users are able to browse books by title, author, keyword, or publish year.
- Instructors are abe to assign certain books to certain students with a due date.
- Users can request adding new books to the library, and the librarians can review this requests and either accept or reject them
- If a book exists in the system and is not already borrowed by another user, users can request to borrow these books. These requests can either be accepted with a due date or rejected by a librarian
- Users are able to reserve books if they are already borrowed. These requests can either be accepted or rejected by a librarian
- Users are able to return books after they are done reading
- Borrowed, returned, and reserved books stay in a user's account
- Librarians are able to send notifications to users, mostly regarding their borrowed books
- These notifications are visible to a student in the **Notifications** page

2. Final E/R Diagram



3. Final List of Tables

Format: **table_name**(<u>primary_key</u> data_type, *foreign_key* data_type, field data_type)

assignment(<u>assignment_id</u> bigint, due_date bigint, weight varchar(5000), assigned_book_id bigint, *student_id* bigint, *instructor_id* bigint)
book(<u>book_id</u> bigint, author varchar(500), genre varchar(500), isbn varchar(500), issn varchar(500), language varchar(500), published_year int, publisher varchar(500), title varchar(500))
book_rating(<u>rating_id</u> bigint, comment varchar(500), *rater_id* bigint, rating_date bigint, score int, *book_id* bigint)
book_request(<u>request_id</u> bigint, author varchar(500), genre varchar(500), isbn varchar(500), issn varchar(500), language varchar(500), published_year int, publisher varchar(500), request_date bigint, title varchar(500), *requester_id* bigint)
borrow_record(<u>borrow_id</u> bigint, borrow_date bigint, due_date bigint, return_date bigint, *approver_id* bigint, *book_id* bigint, *requester_id* bigint)

notification(notification_id bigint, notification_date bigint, text varchar(5000), *receiver_id* bigint, *sender_id* bigint)
reservation_record(reservation_id, request_date, *approver_id*, *book_id*, *requester_id*)
student(department varchar(50), fines double(5,2), year int, user_id bigint)
user(user_id bigint, bilkent_id varchar(8), email varchar(100), first_name varchar(20), last_name varchar(20), password varchar(100), role int)
instructor(department varchar(50), fines double(5,2), sections varchar(500), user_id bigint)
librarian(years_of_experience int, user_id bigint)

4. Implementation Details

We used Java Spring Framework to develop our project. Project's front-end pages were made with HTML and CSS, while the back-end code was written in Java. The database was developed with MySQL. Tables were created using Java Spring's Entity objects, the column names, primary keys, foreign keys and constraints were specified there. Other queries (insert, delete, create) were created with SQL codes.

In order to connect front-end to back-end, Thymeleaf, a template engine that provides Java Spring integration was used. In Java Spring controllers, data of the objects (book, assignment etc.) were retrieved through SQL queries. The data was then directed to the HTML page using controllers and shown in HTML data using Thymeleaf.

Example Problems:
In order to implement book borrow system, we had to check several conditions in the book information page. First we checked whether the user already sent a borrow request. If no such request existed, then we proceeded with checking if another user send a borrow request and got approved. If this condition also did not hold, we allowed the user to sent a borrow request. If borrow operation is not available, user should be able to send a reservation(hold) request. It would be ridiculous to have borrow and reserve buttons existing on the page at the same time. So we used Thymeleaf's thy:if condition for button tags to hide/show mentioned buttons. Thy:if condition's variable were sent from the Java Spring's controller class BookController.

Navigation bar was also implemented with a similar solution. As different user types(student, librarian, instructor) had different functions (example: librarian can access the page for adding books), we used thy:if to design 3 different navigation bars on the same html page and showed one of them depending on user's role. The user role was sentthrough Java Spring's controller class BookController.

Group Member Responsibilities:

Ali Doğaç Urkaya: Created SQL queries for some objects and designed tables. Edited website pages in order to integrate them with the back-end (Wrote Thymeleaf code). Did some minor html for thymeleaf.

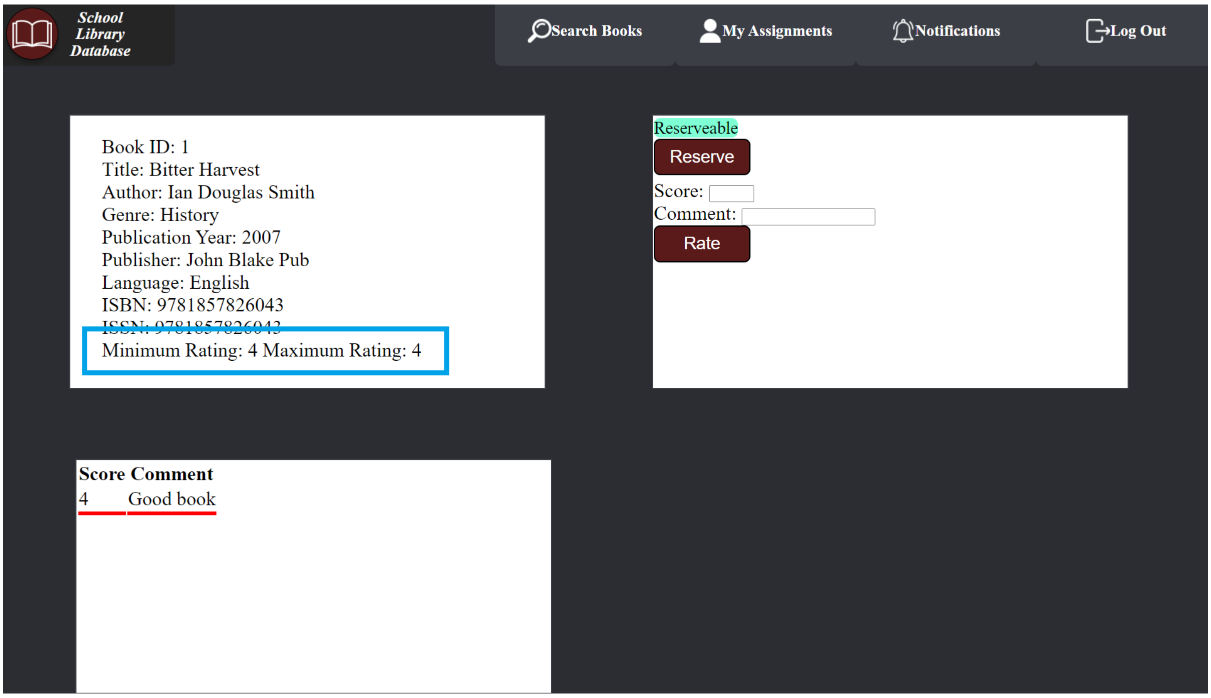
Serhat Gürgenyatağı: Created the essential styling (CSS) for the entire website. Created the 3 different navigation bars specific to the roles (Student, Instructor, Librarian), both CSS and HTML. Mostly worked on front-end and created various pages of the system

Özgür Abi: Created SQL queries for most of the objects and designed tables. Edited website pages in order to integrate them with the back-end (Wrote Thymeleaf code). Created Java Spring entities and controllers.

Jankat Berslan Dinçer: Took part in design of the database systems. Worked in the front-end implementation of some pages with HTML and CSS.

5. Advanced Database Features

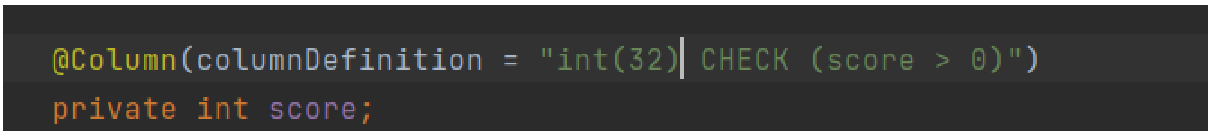
Below is an example (highlighted in the blue box) of an instance where we used complex SQL queries (MAX, MIN).



Book ID: 1
Title: Bitter Harvest
Author: Ian Douglas Smith
Genre: History
Publication Year: 2007
Publisher: John Blake Pub
Language: English
ISBN: 9781857826043
ISSN: 9781857826043
Minimum Rating: 4 Maximum Rating: 4

Score	Comment
4	Good book

We also used constraints in our project. Book rating score can not be negative (Users can't give negative ratings to books).

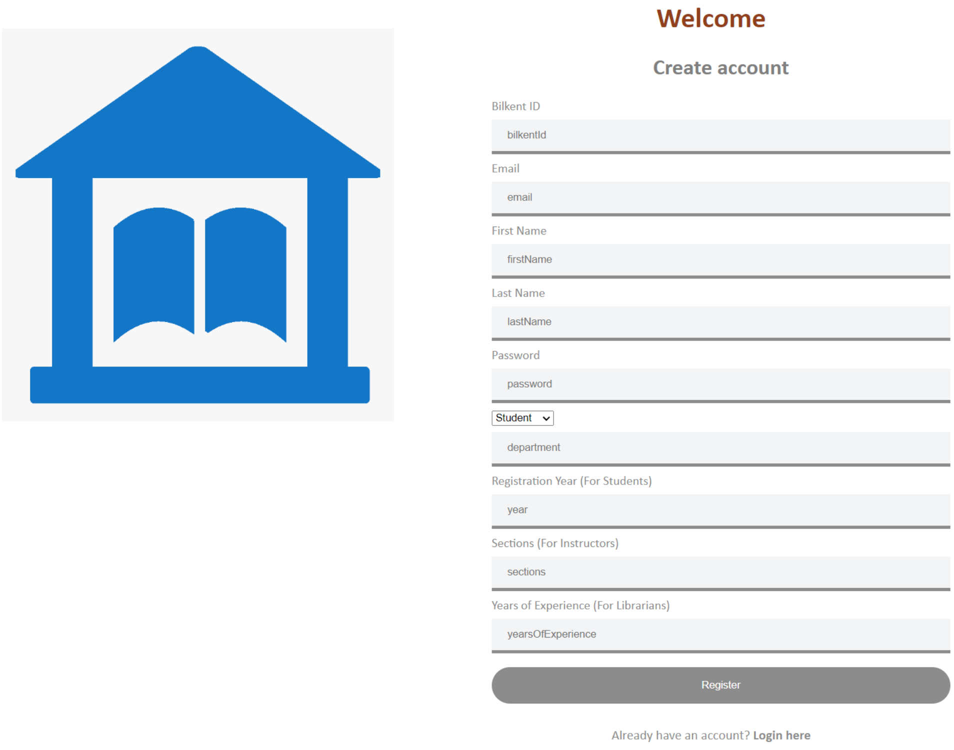


```
@Column(columnDefinition = "int(32) CHECK (score > 0)")
private int score;
```

6. User's Manual

6.1. Login Screen

The application will have 3 different types of users. These are students, instructors, and librarians. The librarians will log into the system using their IDs given to them by the application's administors.



Welcome

Create account

Bilkent ID

bilkentId

Email

email

First Name

firstName

Last Name

lastName

Password

password

Student ⌄

department

Registration Year (For Students)

year

Sections (For Instructors)

sections

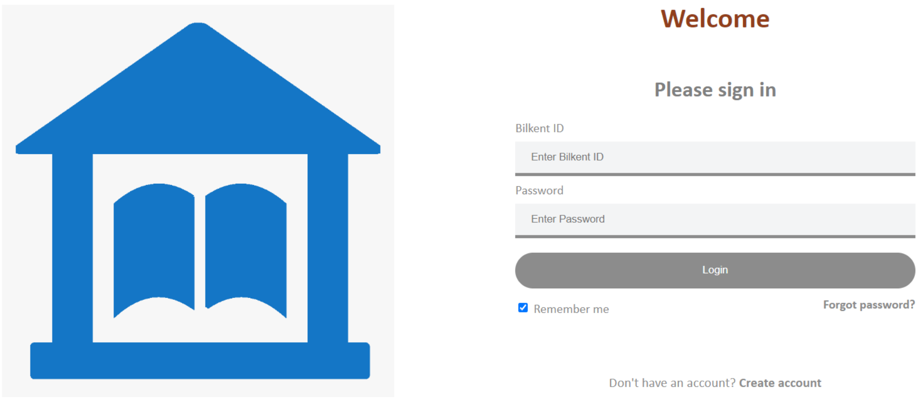
Years of Experience (For Librarians)

yearsOfExperience

Register

Already have an account? **Login here**

They will be able to create accounts for the students and instructors and they will be responsible for creating all accounts and sending their information to the users.



Welcome

Please sign in

Bilkent ID

Enter Bilkent ID

Password

Enter Password

Login

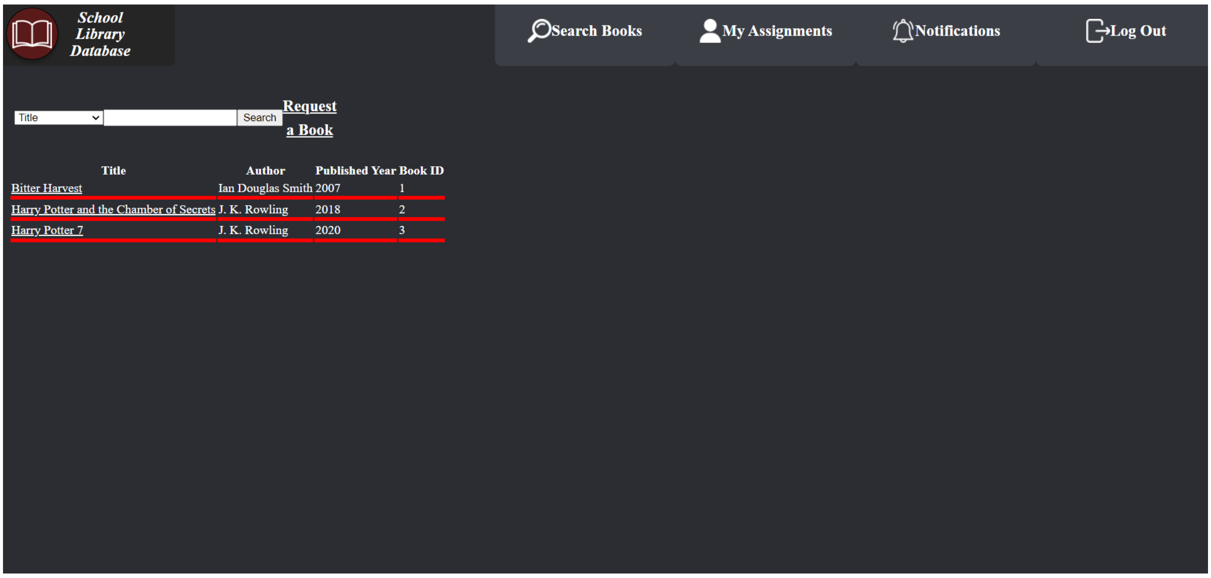
☑ Remember me

Forgot password?

Don't have an account? **Create account**

Through the login screen users will enter their credentials and click on the login button to access the system. All 3 types of users will log in through the same system.

6.2. Book Search Screen

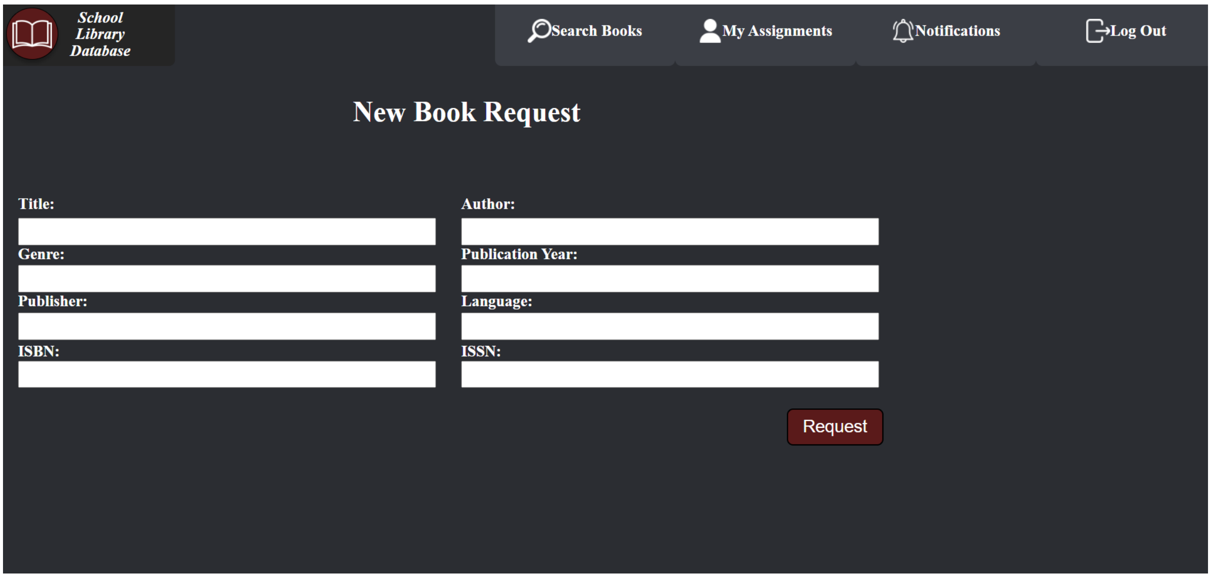


Title	Author	Published Year	Book ID
Bitter Harvest	Ian Douglas Smith	2007	1
Harry Potter and the Chamber of Secrets	J. K. Rowling	2018	2
Harry Potter 7	J. K. Rowling	2020	3

Students and instructors are automatically directed to this screen when they first log in. Users can also return to this screen through the navbar. The user sees a list of all books in the library database.Users can search the library database for specific books using the search bar. They can filter the search bar for book titles, authors, or keywords. They set their filters and enter the related keywords into the search bar. They can then click on “Search” to make a search and find any books they are looking for. They can then click on a book to see detailed information about the book, including reviews left by other users, and either make a request to borrow the book, put the book on hold, or leave a review.

If the book the user was looking for does not exist within the system, they can click on “Request a Book” to make a formal request for the librarians to add the book to the library system. This request will be reviewed and either accepted or rejected by librarians.

6.3. Book Request Screen



New Book Request

Title:

Author:

Genre:

Publication Year:

Publisher:

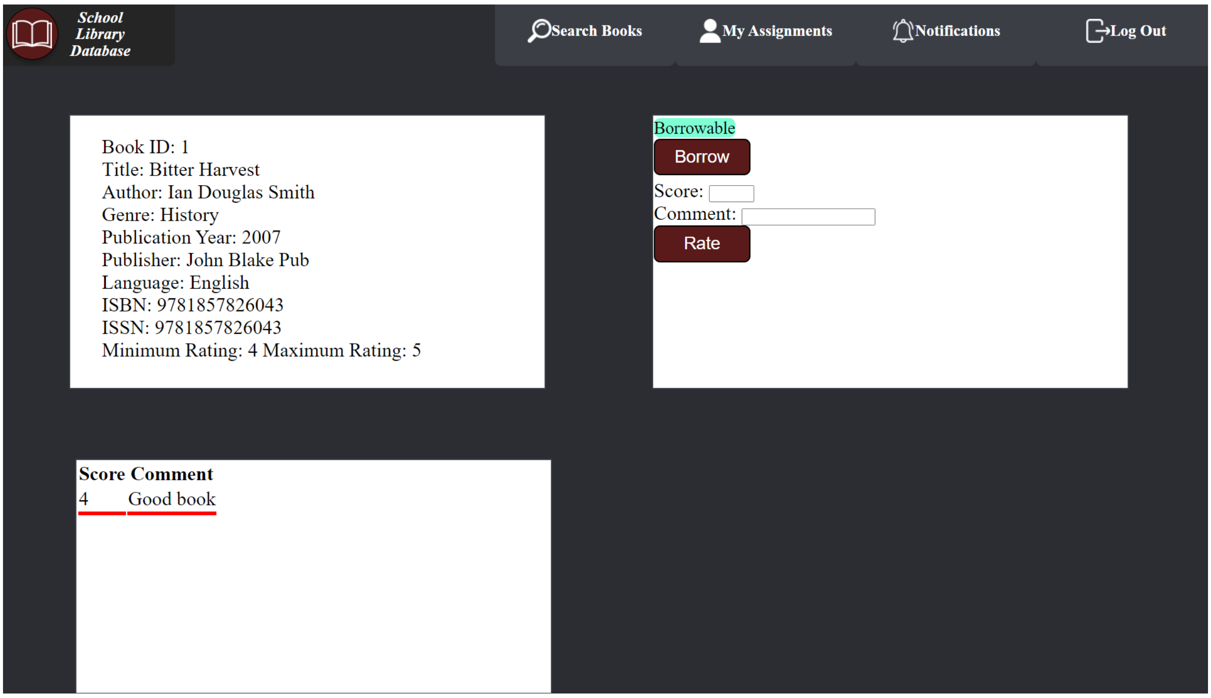
Language:

ISBN:

ISSN:

When a user presses the “Request a Book” button they are taken to the book request screen. The user then enters the necessary information about the book that they are requesting so that the librarians can easily find the specific book the user is looking for. Then the user clicks on the “Request” button and sends their request. The request will later be checked and approved or rejected by the librarian.

6.4. Book Screen for Ratings and Reservations



Book ID: 1
Title: Bitter Harvest
Author: Ian Douglas Smith
Genre: History
Publication Year: 2007
Publisher: John Blake Pub
Language: English
ISBN: 9781857826043
ISSN: 9781857826043
Minimum Rating: 4 Maximum Rating: 5

Borrowable

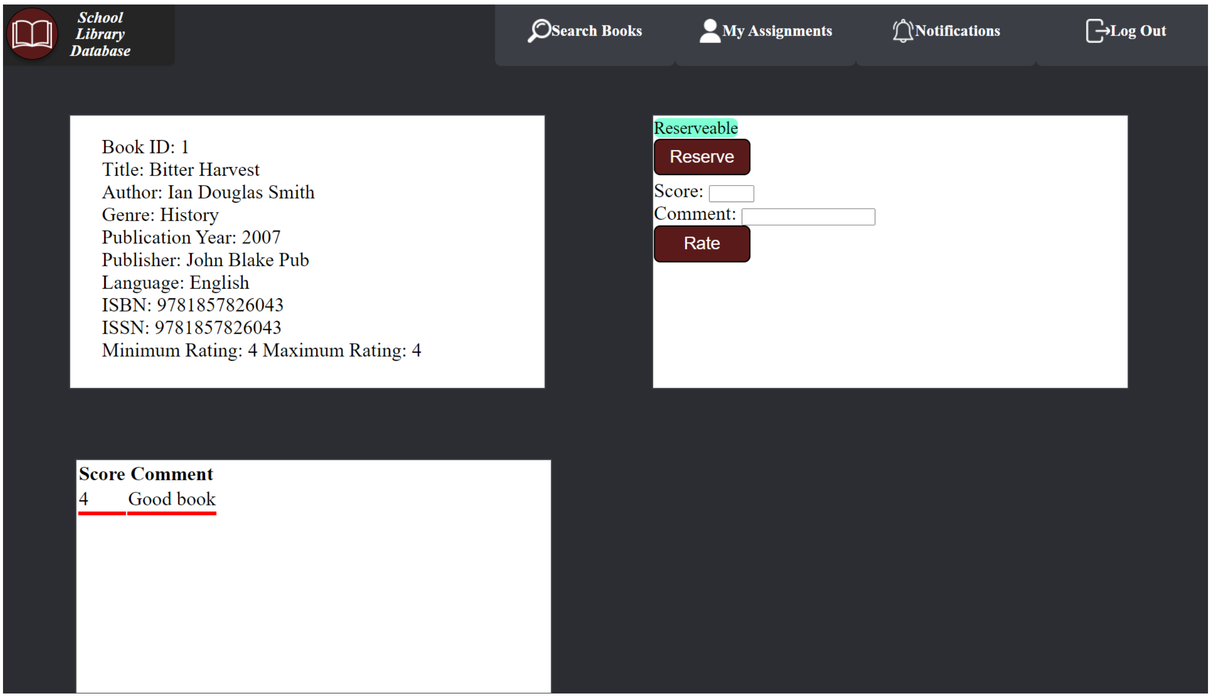
Score	Comment
4	Good book

After clicking on a book the user will be taken to a page where detailed information about the book can be seen.

On the top left side of the screen information about the book is displayed.

On the bottom left side the users can read reviews of the book written by other users.

On the right side of the screen users can make requests to borrow or reserve books. If a book is currently in the system and not in possession of another user, the book is displayed by the system as being “Borrowable”. The user can click on the “Borrow” button to make a borrow request which will later be approved or rejected by the librarian.



Book ID: 1
Title: Bitter Harvest
Author: Ian Douglas Smith
Genre: History
Publication Year: 2007
Publisher: John Blake Pub
Language: English
ISBN: 9781857826043
ISSN: 9781857826043
Minimum Rating: 4 Maximum Rating: 4

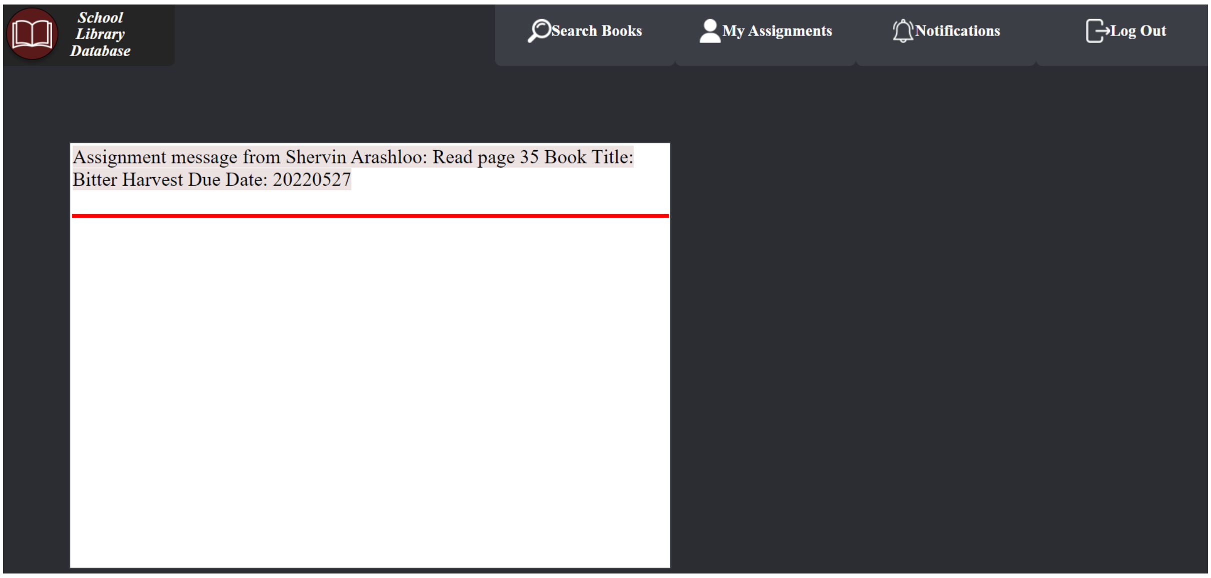
Reserveable

Score	Comment
4	Good book

If the book is currently in possession of another user, the system displays the book as being “Reservable”. The user can then click on the “Reserve” button to make a reservation request and put the book on hold which will later be approved or rejected by the librarian.

Users can also use the “Score” and “Comment” bars to enter a score out of 5 and a comment, and leave a review for the book for other users to read.

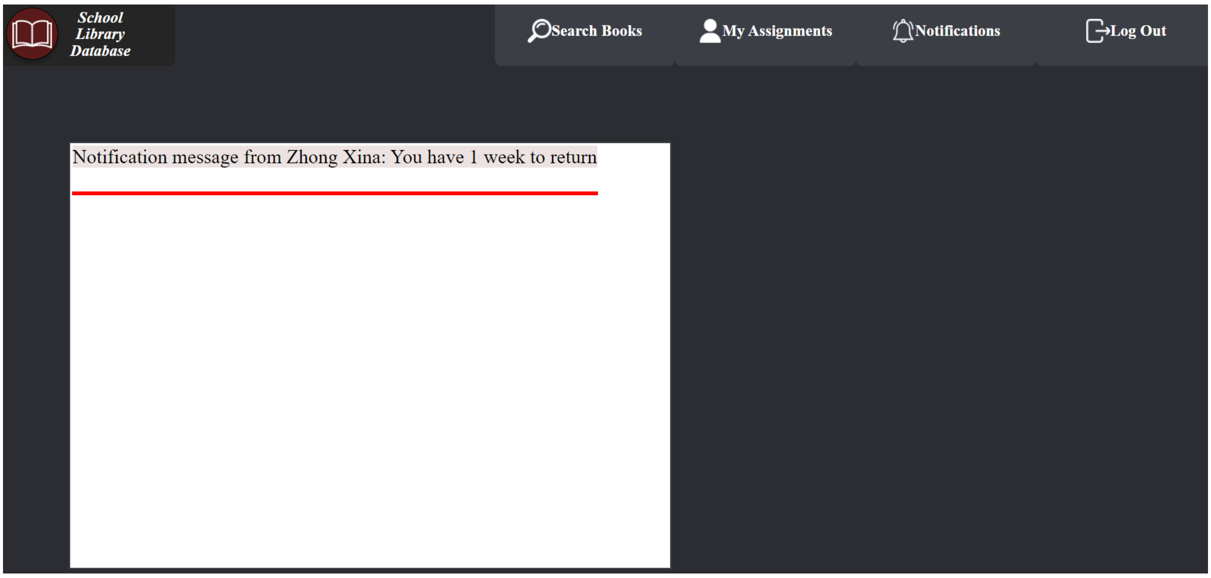
6.5. Assignments Screen



Assignment message from Shervin Arashloo: Read page 35 Book Title: Bitter Harvest Due Date: 20220527

By navigating to the “My Assignments” tab on the navigation bar, users can access the assignments screen. There, they can see books that have been assigned to them by their instructors.

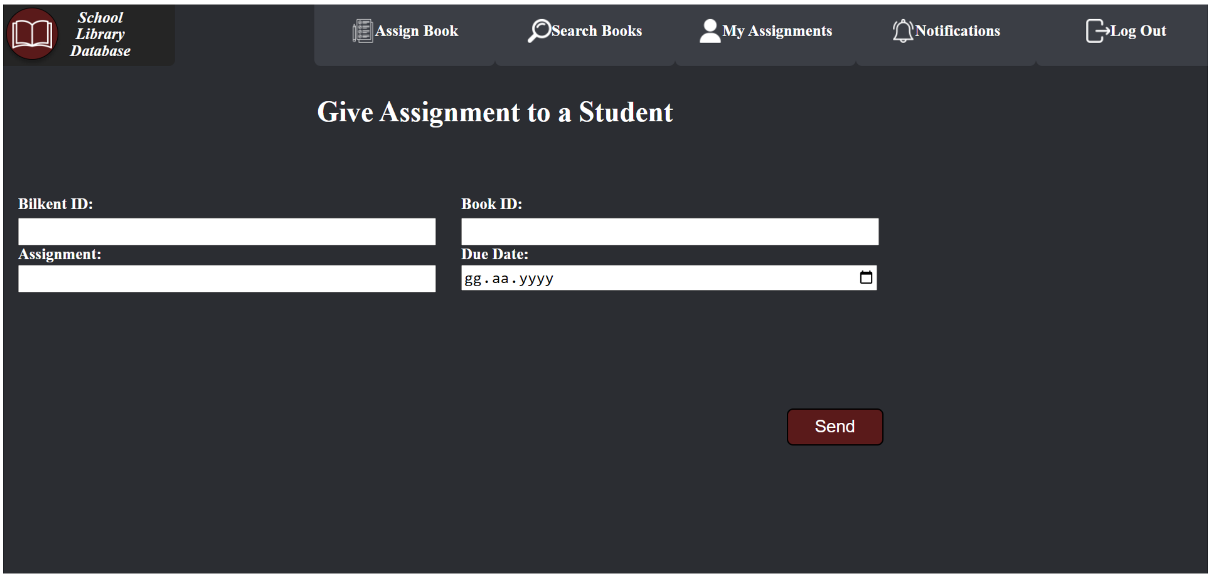
6.6. Notifications Screen



Notification message from Zhong Xina: You have 1 week to return

By navigating to the “Notifications” tab on the navigation bar, users can access the notifications screen. There, they can see notifications sent to them by the librarians on upcoming due dates or books in their possession that are due.

6.7. Give Assignment Screen for Instructors



Give Assignment to a Student

Bilkent ID:

Book ID:

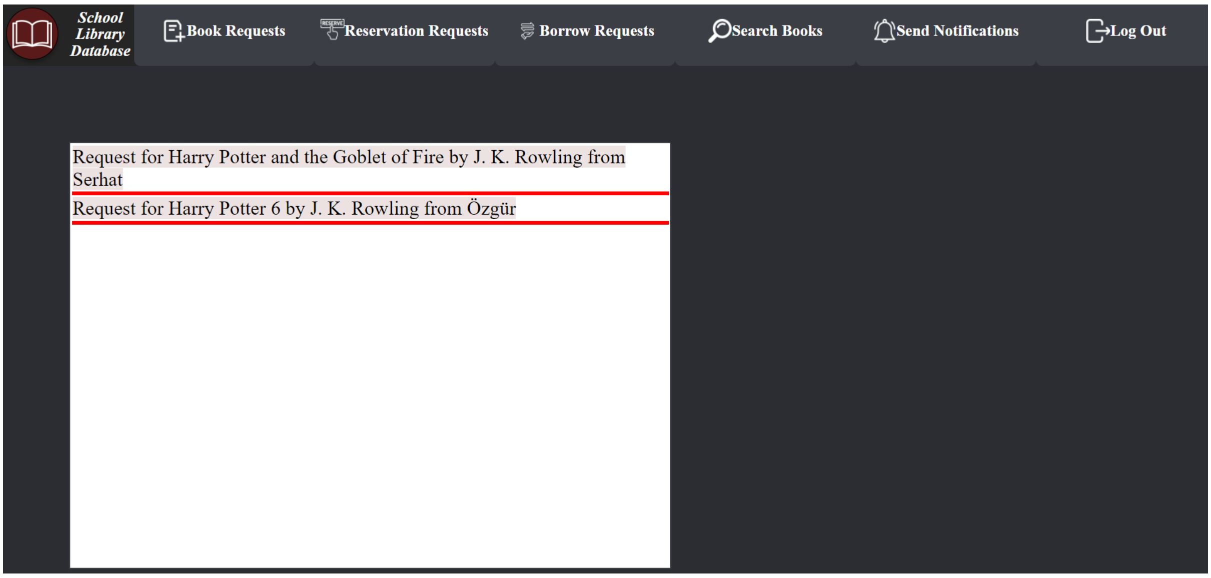
Assignment:

Due Date: gg.aa.yyyy

The instructors can navigate to the “Assign Book” tab on the navigation bar to get to the assign book screen. There, they can enter the ID of the student who is being assigned the book, the ID of the book being assigned (this ID is visible on the book information screen), the due date of the assignment, and any extra comments they might be willing to make on the assignment (Such as which pages are assigned).

Instructors are also able to access all previously mentioned pages and use the library system the same way students do.

6.8. Book Requests Screen for Librarians

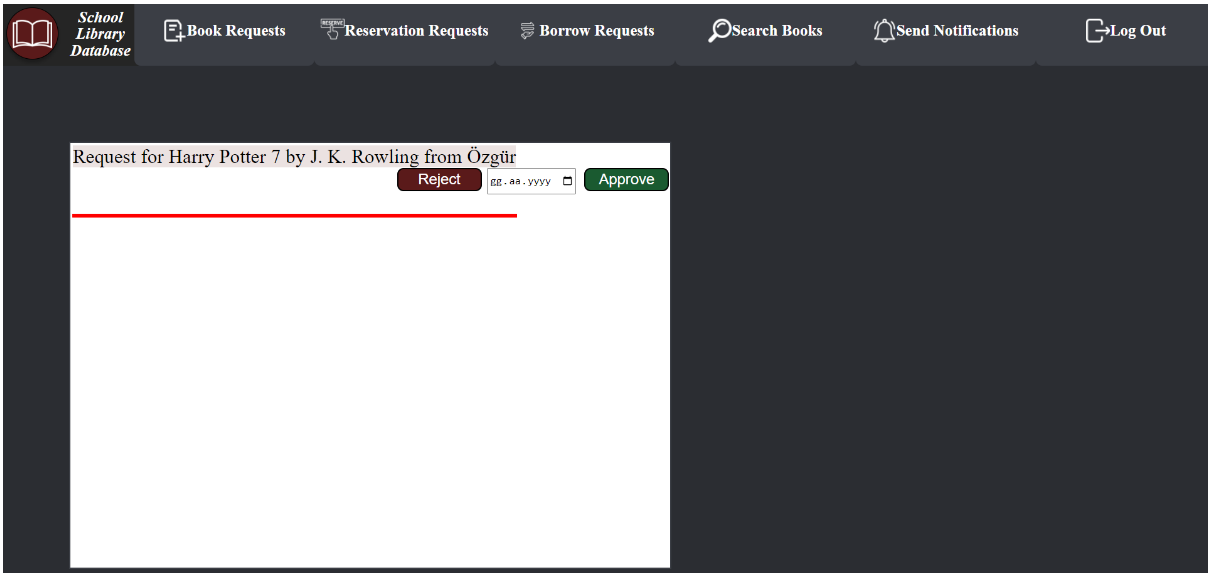


Request for Harry Potter and the Goblet of Fire by J. K. Rowling from Serhat

Request for Harry Potter 6 by J. K. Rowling from Özgür

The librarian can access this screen from the navigation bar by pressing the “Book Requests” button. When a student requests for a new book to be added to the system, the request has to be approved by the librarian. Through this screen librarians can see all requests for new books to be added, and they can click on the respective buttons to either reject or approve this request.

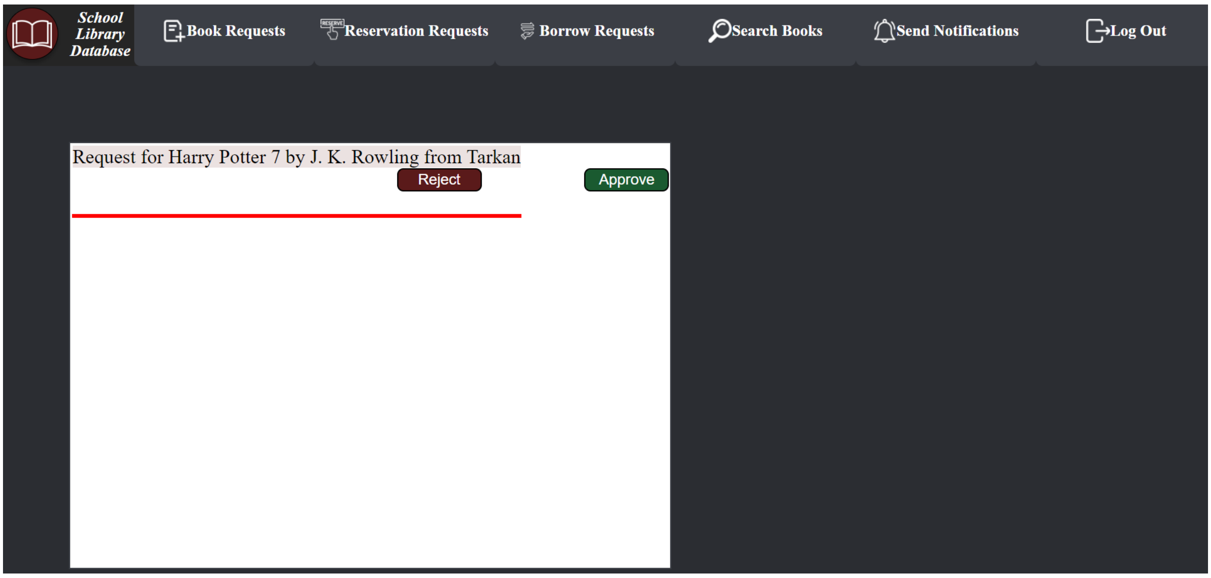
6.9. Borrow Requests Screen for Librarians



Request for Harry Potter 7 by J. K. Rowling from Özgür

The librarian can access this screen from the navigation bar by pressing the “Borrow Requests” button. The librarian can see a list of all borrow requests made by the users. For each request, the librarian can then choose to either reject the request or set a due date for the book and approve the request through the respective buttons.

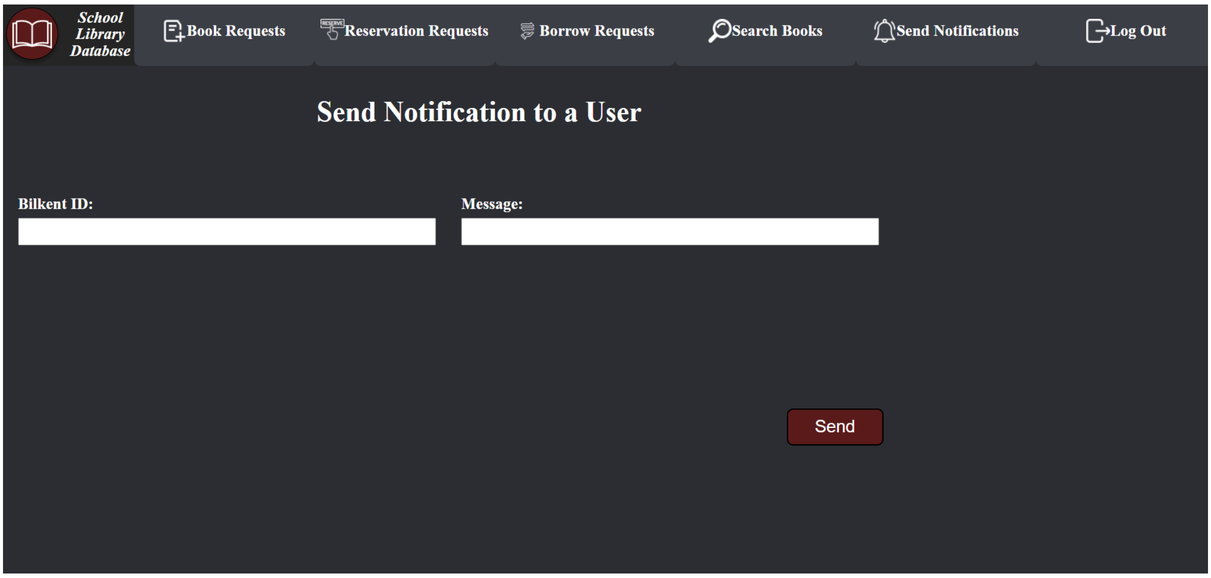
6.10. Reservation Requests Screen for Librarians



Request for Harry Potter 7 by J. K. Rowling from Tarkan

The librarian can access this screen from the navigation bar by pressing the “Reservation Requests” button. The librarian can see a list of all reservation requests made by the users. For each request, the librarian can then choose to either reject the request or approve the request through the respective buttons.

6.11. Send Notifications Screen for Librarians



The librarian can access this screen from the navigation bar by pressing the “Send Notification” button. The librarian can enter the ID of a user and a message into the text boxes and click on the “Send” button to send a notification to a user. These notifications can be used to warn users of upcoming due dates or warn them about past due dates of books they are in possession of.