CS353 Database Systems Project Final Report

School Library Database System

Spring 2022

Group 23

Elifnur Alsaç - 21601098 - Section 2

Esad İsmail Tök - 21801679 - Section 2

Mohamed Wasim Mohamed Akram - 21801103 - Section 2

Application System Description

In this project it is aimed to produce a Library management system with 3 different users which are librarian, instructor, and student. Each user type has different authorizations and the type of the application user in the program is chosen in the login screen. In case of a login or registry failure, relevant error messages are sent to the users.

A librarian can create and delete users, add and delete books in the library, approve or disapprove the borrowing requests of the books for the student type of users.

Student type of user can hold books and after the approval by the librarian, they can borrow the books in the library. Name, author, and due dates of each book can be seen in the main page of the application.

Instructor user can assign books to the student users in the instructor dashboard however only the frontend is implemented for this feature and not the backend.

Student users can view their assigned books by the instructors and they can submit their assigned homework by the submit button but only the frontend of this feature is implemented but not the backend.

Librarian users can generate warning messages and send those messages to other users and other user types can view those warning messages in their dashboards.

There is an alert system in the application that can be seen by all types of users in the application that aims to inform users about the recent news and notifications.

There is an extra feature that we have planned to add to our application but we could not connect the front end of the feature to the backend of the application due to the time limitation. This feature allows students to make reviews and comments on the books that they have borrowed using the review button which is placed in the section of each borrowed book in students borrowed books list.

Final E/R Diagram

The final E/R diagram that we are using is the same as the one that we used in our design report. *Figure 1* represents the final E/R diagram of our project.

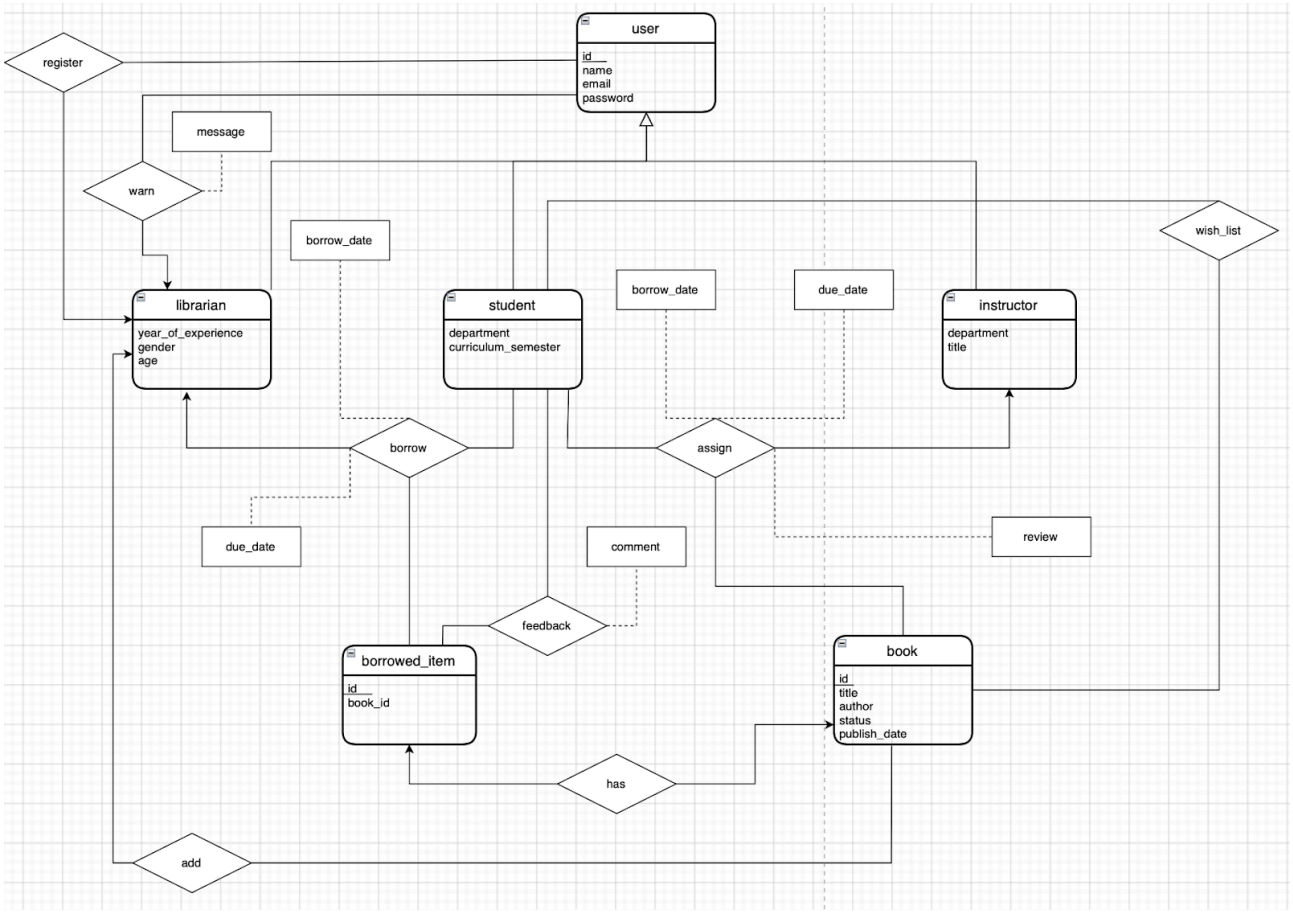

Figure 1: Final E/R Diagram

Final List of Tables

We used the same relation schemas in our project after the design report so we need no change in the list of tables after the design report.

user(id, name, email, password)

librarian(id, year_of_experience, gender, age)

student(id, department, curriculum_semester)

instructor(id, department, title)

borrowed_item(id, book_id)
Foreign Key book_id references book(id)

book(id, title, author, status, publish_date)

register(user_id, librarian_id)
Foreign Key user_id references user(id)
Foreign Key librarian_id references librarian(id)

warn(user_id, librarian_id, message)
Foreign Key user_id references user(id)
Foreign Key librarian_id references librarian(id)

add(book_id, librarian_id)
Foreign Key book_id references book(id)
Foreign Key librarian_id references librarian(id)

has(borrowed_item_id, book_id)
Foreign Key borrowed_item_id references borroved_item(id)
Foreign Key book_id references book(id)

wish_list(student_id, book_id)
Foreign Key student_id references student(id)
Foreign Key book_id references book(id)

feedback(student_id, borrowed_item_id, comment)
Foreign Key student_id references student(id)
Foreign Key borrowed_item_id references borroved_item(id)

borrow(student_id, borrowed_item_id, librarian_id, borrow_date, due_date)
Foreign Key student_id references student(id)
Foreign Key borrowed_item_id references borroved_item(id)

assign(student_id, book_id, borrow_date, due_date, review)
Foreign Key student_id references student(id)
Foreign Key book_id references book(id)

Implementation Details

In the frontend of our application we used HTML5, CSS, and JavaScript as the technologies. We have written the codes in VS Code. In the backend of our application we used PHP. No external libraries are used in frontend and backend of the application.

It was a though experience to implement the project generally because of time limitations and the complexity of designing a website from scratch rather than the complexity of designing a database.We have encountered with a couple of technical problems during the implementation process. Frist of all, we often had to use Javascript and php separately but sometimes, we had to use JavaScript inside php code or php inside JavaScript. This becomes more complicated when we want to use a variable from JavaScript in PHP or vice versa. In these kinds of situations, we researched on multiple sites like stackoverflow or geeksforgeeks to find out if they have solutions for these. For using JS inside PHP or vice versa, we can use a script combined with echo. To use variables across platforms, we can use the $_COOKIE[$cookie_name] to pass variable.

Another problem that we faced was when there are inputs in a form and a submit button is used to $_POST these variables, these values are often put into a database if the code is working. However, sometimes, the input is taken but it doesn't add it to the database and there are no errors. If the code has no comments, this can take a really long time to fix since the problem can be across various files. One of the fixes for this is to check the query that inserts the values into the database and see if the names corresponds to the PHP variables and if the order of inserting them is right. If this doesn't work, sometimes the $_SESSION['username'] is stored but session_start() may not be included in the file so there will not be a username to store so the queries won't execute.

Since our group only consists of 3 people it was a bit hard for the group to allocate work and finish the tasks on time. However we still tried to separate the work equally according to the profession areas of the each group member. Wasim contributed most in the implementation process and contributed to backend, frontend, and database fields of the application the functionalities that he contributed were implementing register and login, borrowing books, alert center feature of the application, front ends of the homework assignments given by instructor users and homework submissions by student users. Esad contributed to the database queries written and some database table creations, frontends of the warning messages generated and viewed, addition of the sql queries into the backend of the application. Elifnur contributed to the frontend of the borrowed book review which is the additional feature of the application and some login warnings implementation given by the miss-match of credentials during the login.

Advanced Database Features

We could not add advanced database features in the implementation of our application because we could not get enough knowledge in those topics before the implementation process and also time limitation was also one of the biggest reasons why we could not implement advanced database features.

User's Manual

Registration:

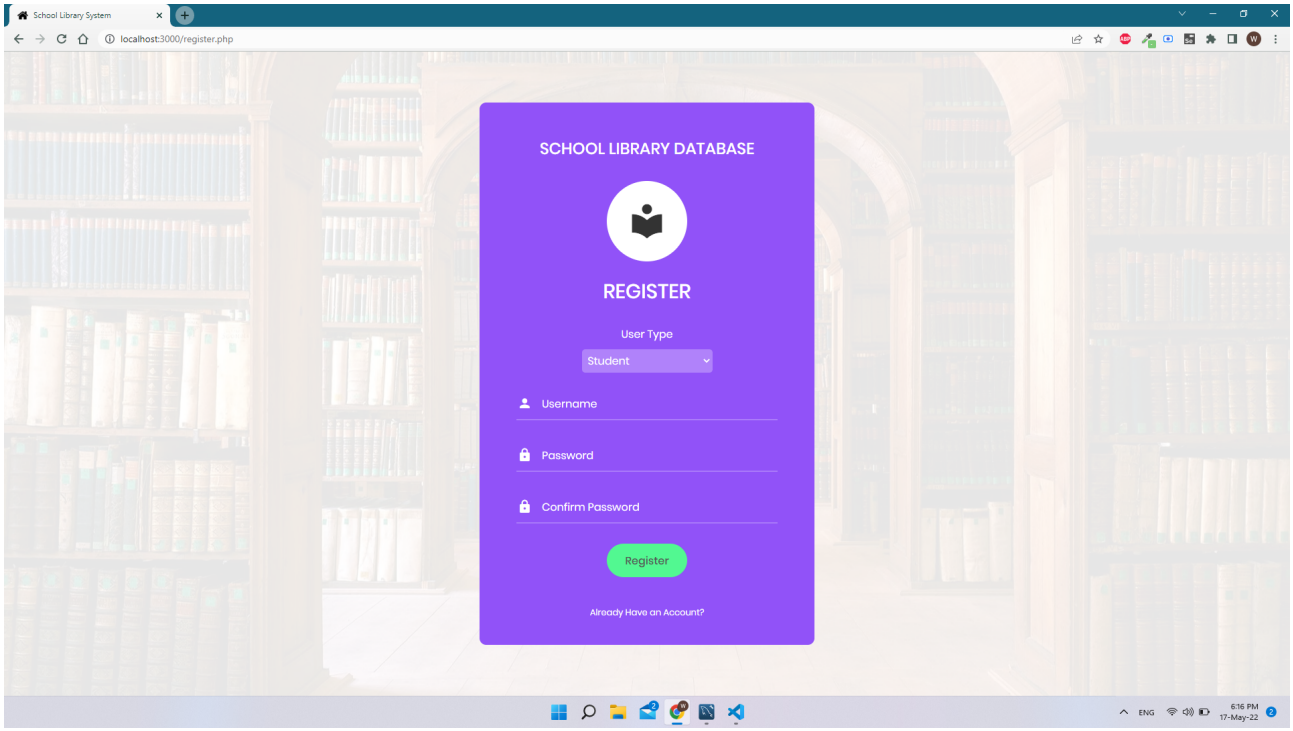

Figure 2: Registration Screen

Figure 2 represents the registration screen of the application in which user provides a username, password, and the confirmation password then presses the registration button to register into the application.

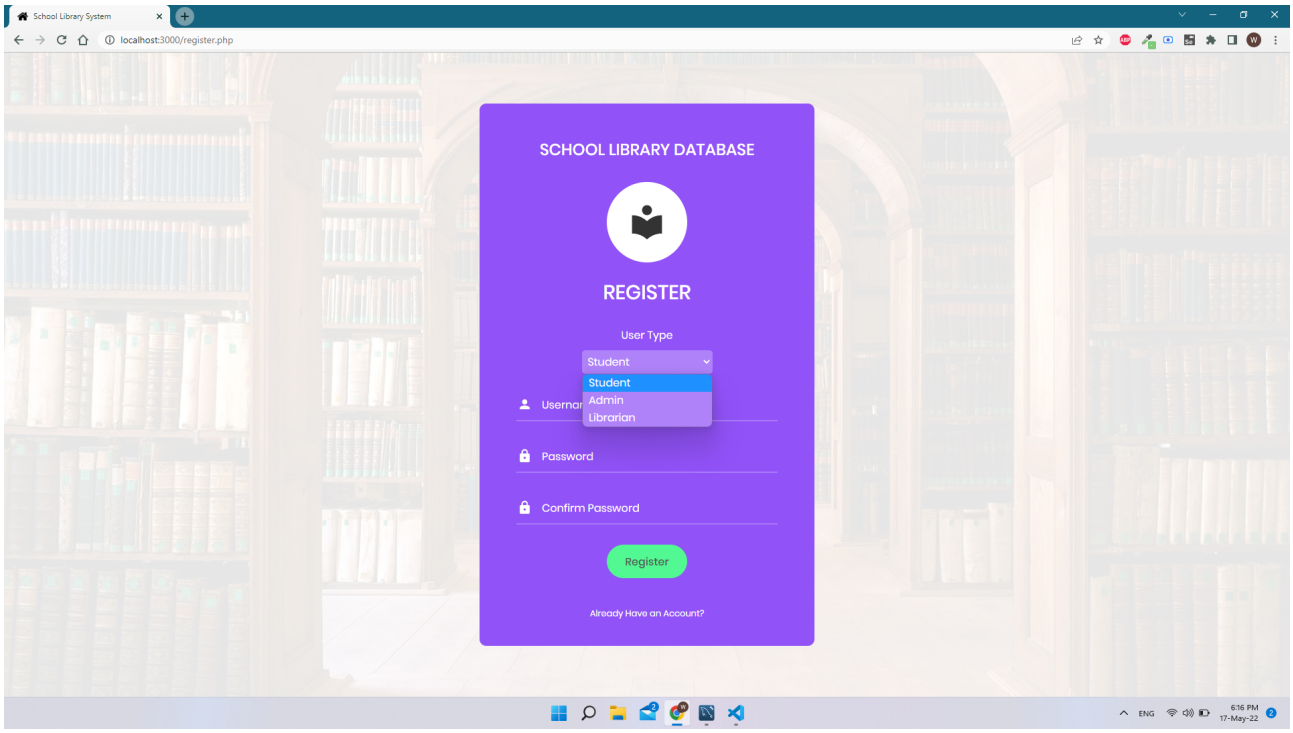

Figure 3: Registration Screen with User Types

Figure 2 represents the registration screen of the application with the user type selections. The registration page of the application is capable of deciding the user type and choosing one of the Student, Admin (Instructor), and Librarian user types using the dropdown menu.

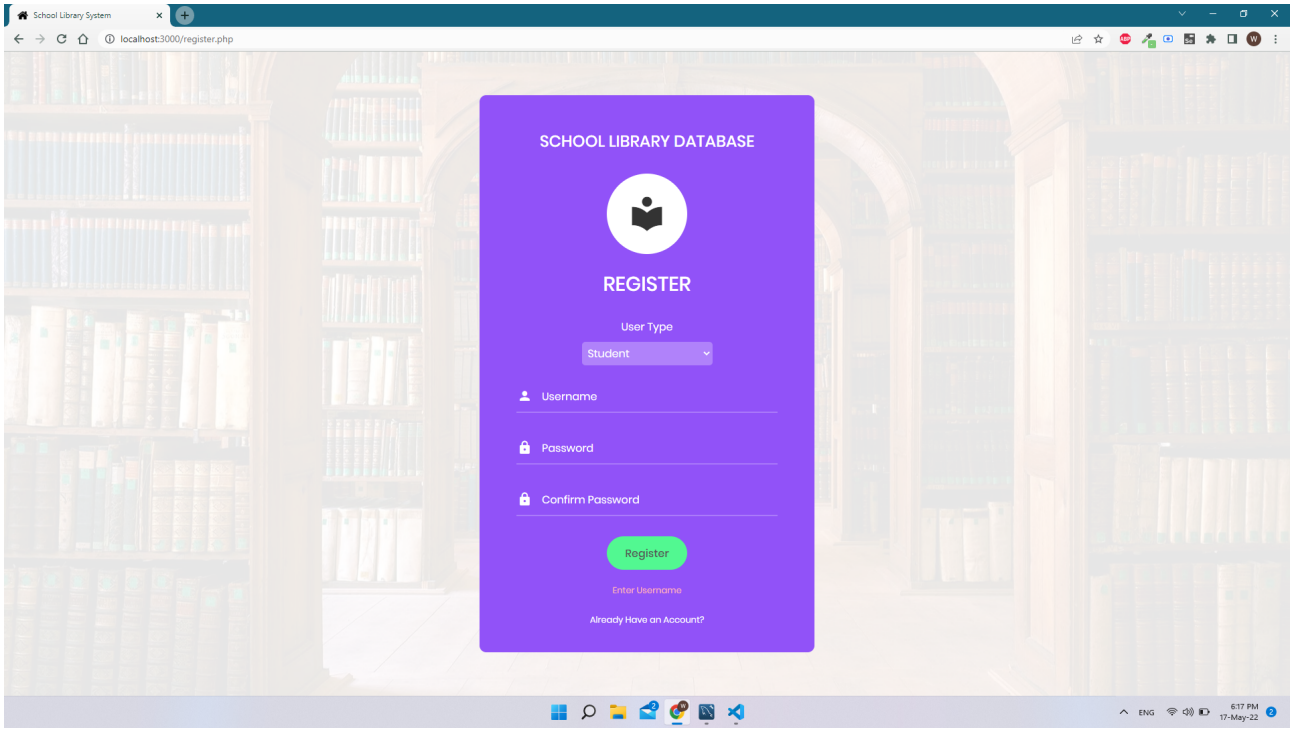

Figure 4: Registration Screen with Error

If the user misses any section in the registration process, the application will give an error saying to fill the missed section. If more than one section is missing, the priority is given to the above section in UI. The error message is displayed just below the register button with the red color.

Login:

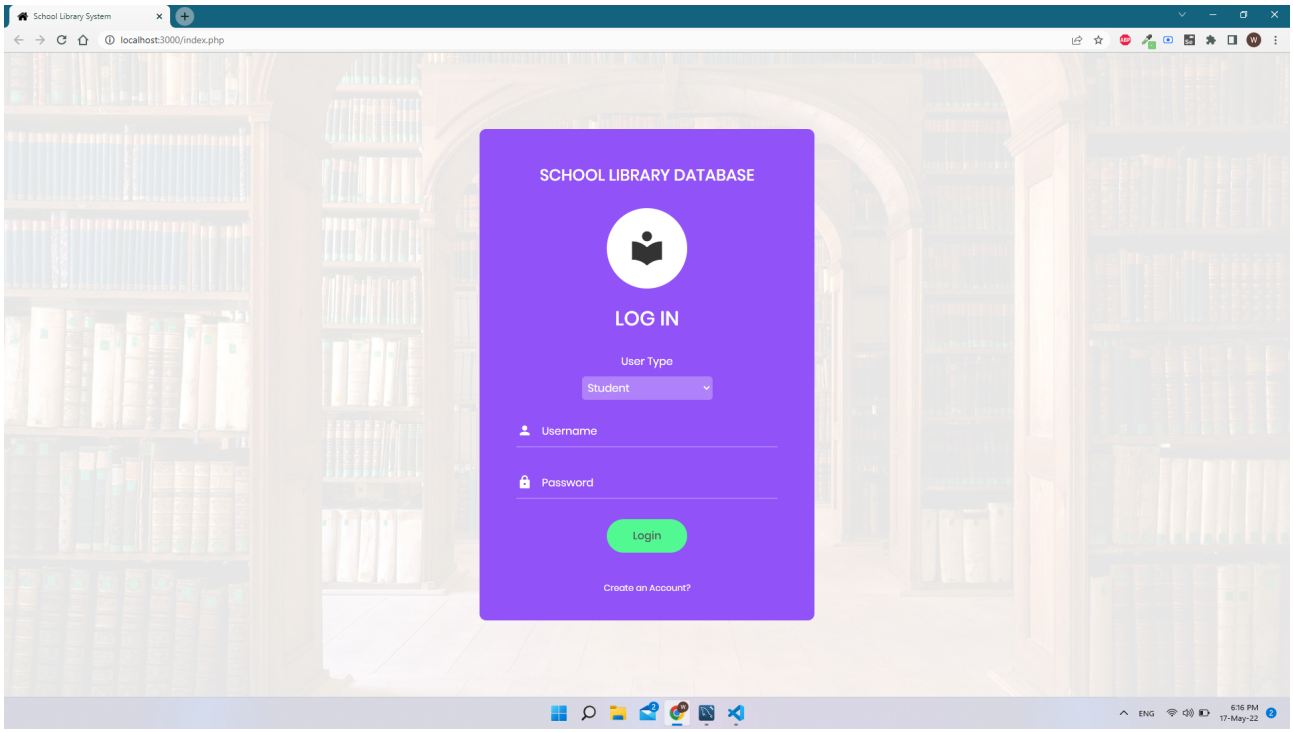

Figure 5: Login Screen

Figure 5 represents the login screen of the application in which user is required to enter his/her username and password that are determined during the registration process. The the user can login to the application by clicking the login button if the validation goes alright.

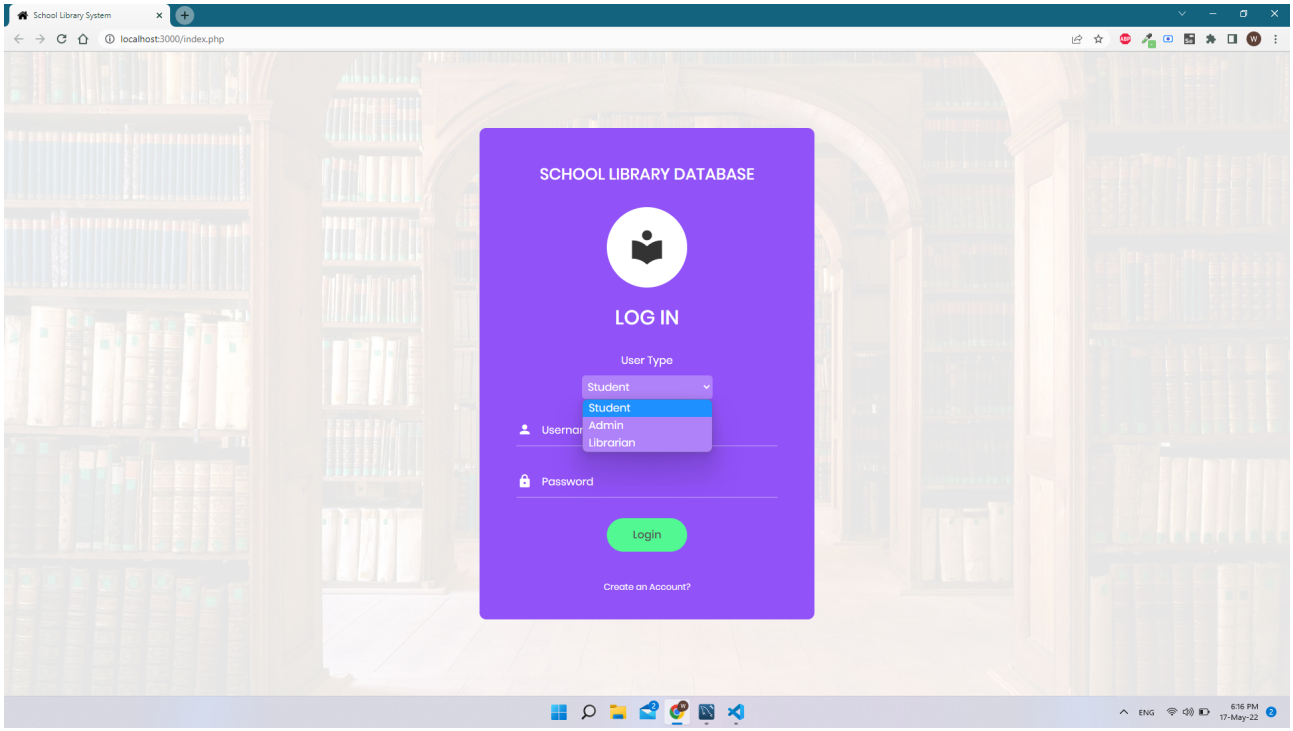

Figure 6: Login Screen with User Types

Login screen also has the capability of choosing the user type and each user should choose his/her user type in the login process. User type can be chosen using the dropdown menu below the user type text.

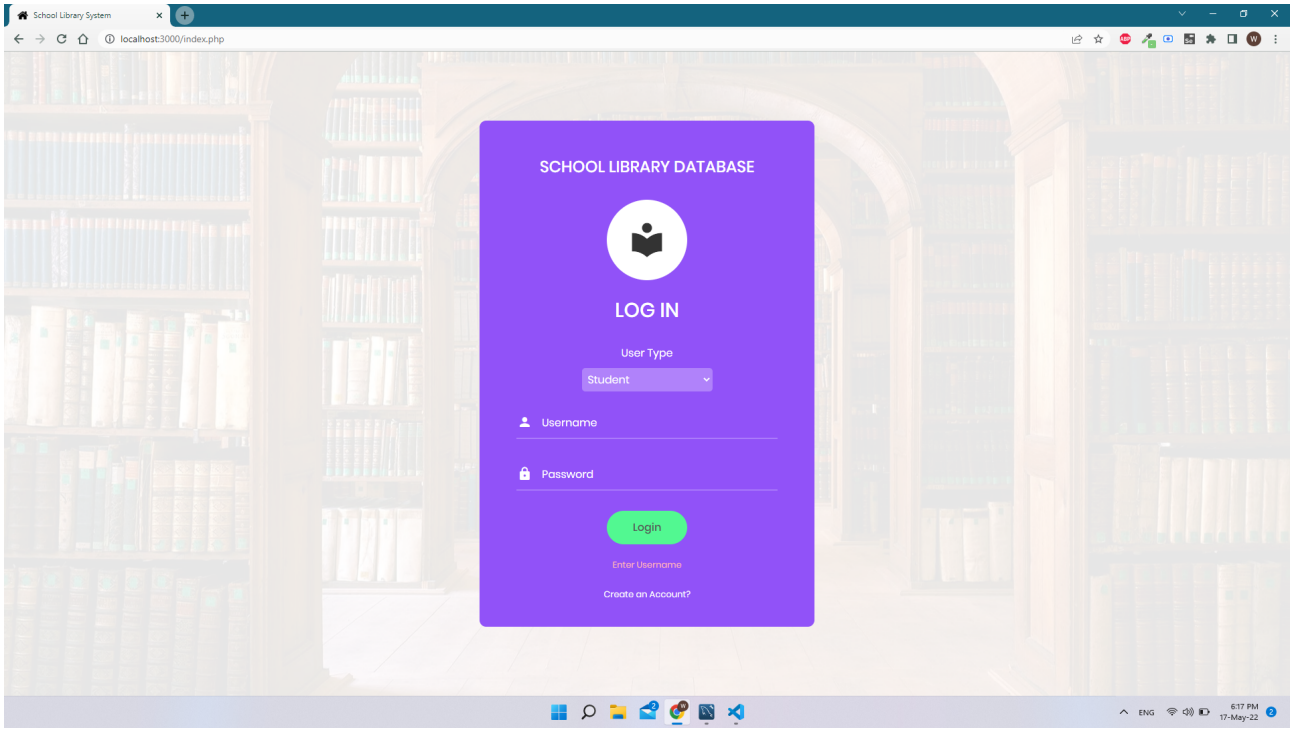

Figure 7: Login Screen with Error

Similar with the registration, login process also requires all the sections to be filled and gives error if any one of those are missing. The application also displays the relevant error messages when there is no such user with the specified username and if the username and passwords do not match.

Library Dashboard:

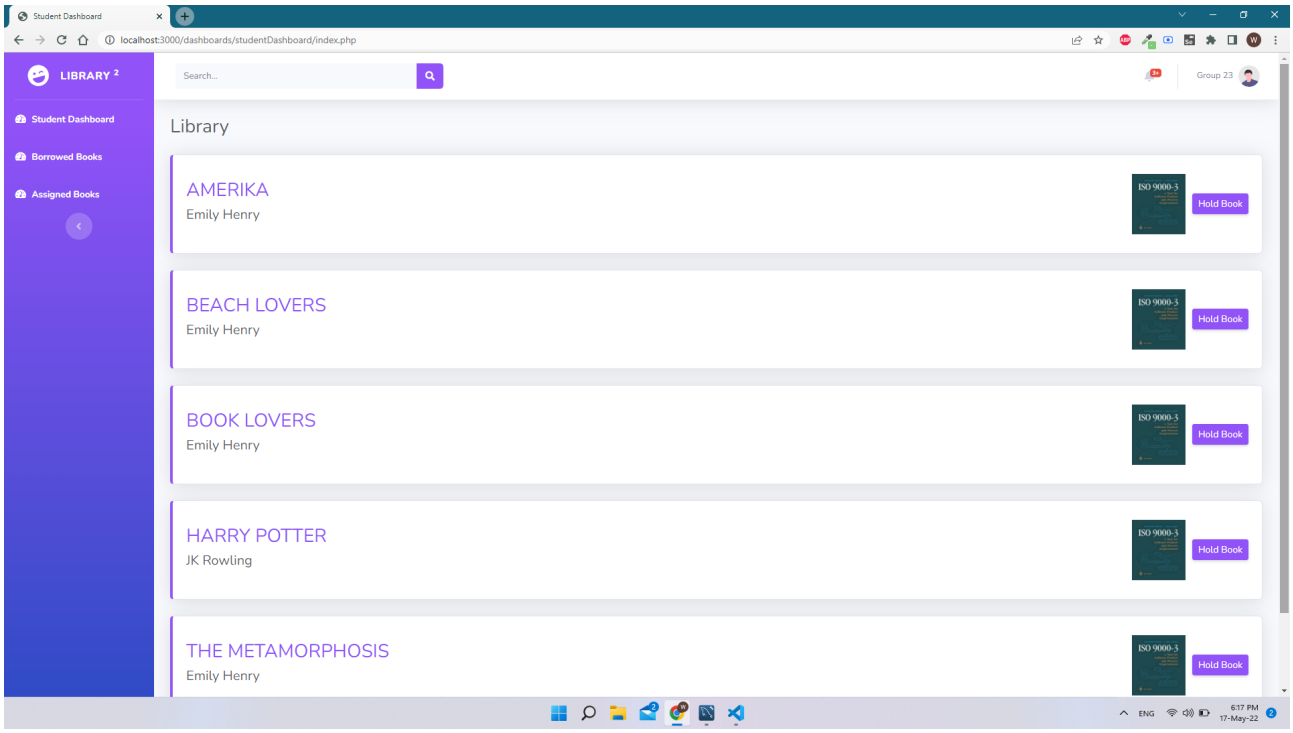

Figure 8: Library Dashboard

Library dashboard includes all the books that are registered to the library and all of the books are sorted in ascending order according to their titles. The title, author and image informations are provided for books. In case a student wants to borrow a book, he/she first needs to hold the book pressing the hold book button next to the book image. Then a request will be sent to the librarian in order to approve or disapprove the request.

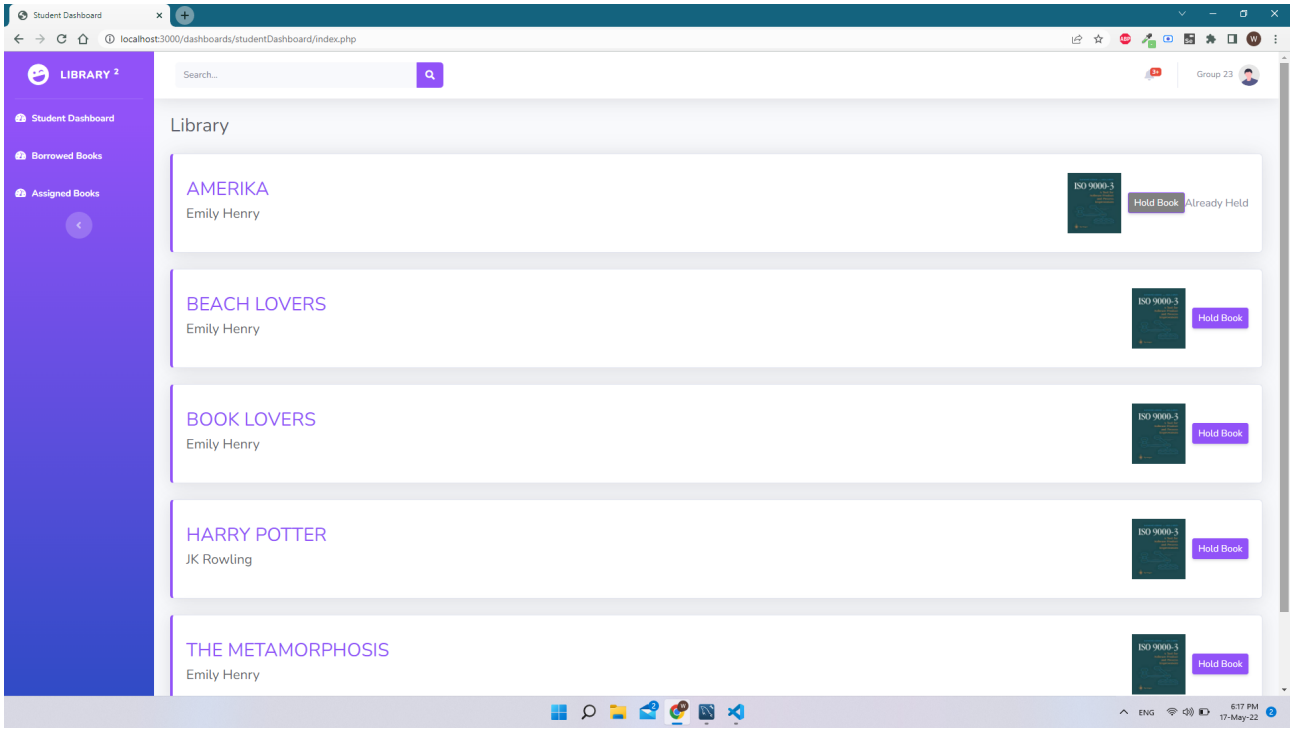

Figure 9: Library Dashboard with Warning

An already held book cannot be held by another user of the library and a warning message saying that the book is already held will be showed to the user.

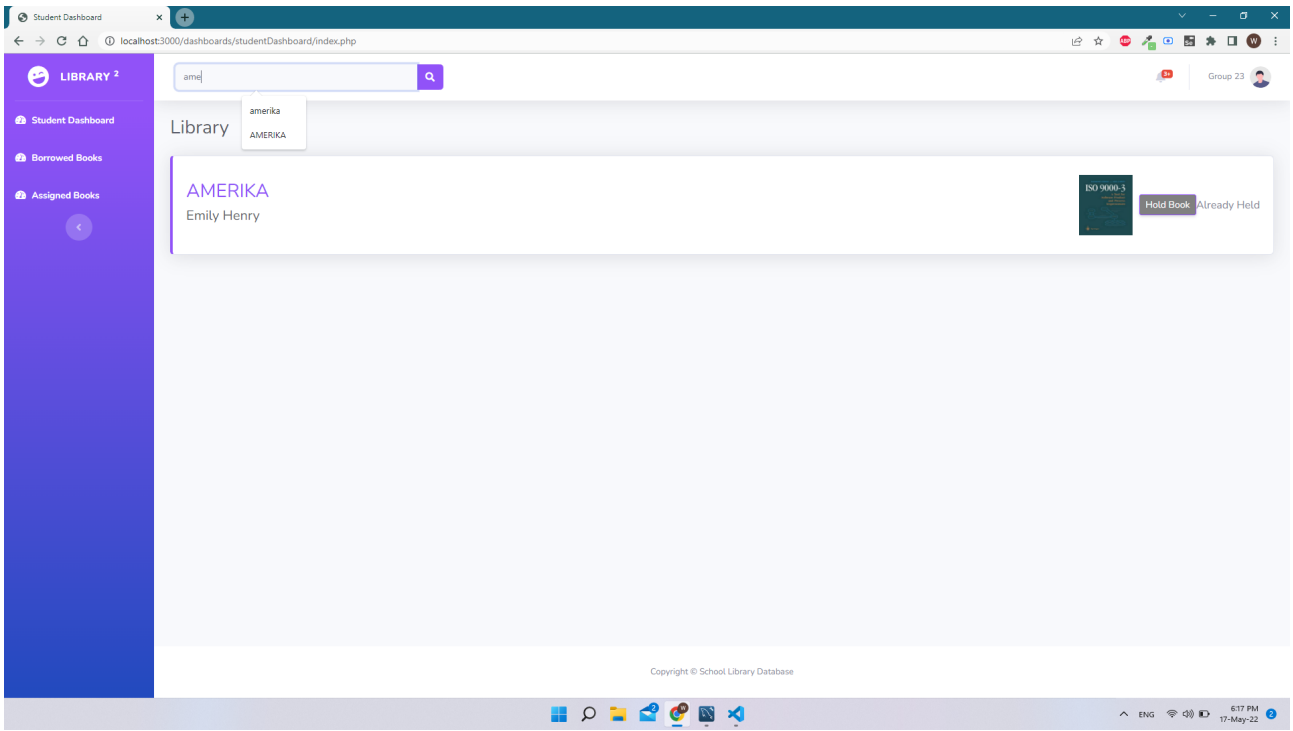

Figure 10: Searching among the Books

Figure 10 shows the search bar at the top left side of the library dashboard in which a user can provide the characters of the title of the book and the matching books are retrieved even though the full title is not provided but only a few characters are provided.

Approval Process:

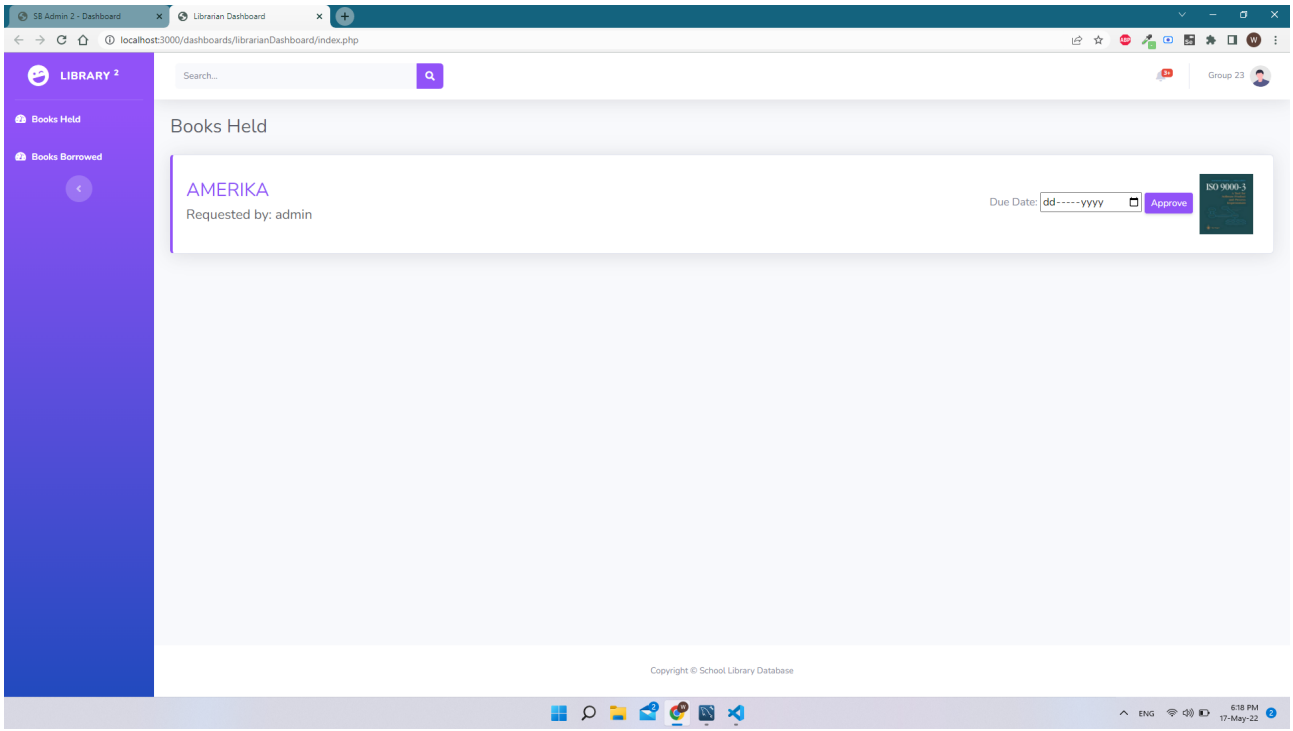

Figure 11: Book Approval Screen for Librarian

After a student holds a book in the library dashboard, an approval request is sent to the librarian user. In his/her dashboard, a librarian can see the books that are waiting for an approval. The name of the book, the name of the user that requests an approval, and the image of the book is provided together with a due date.

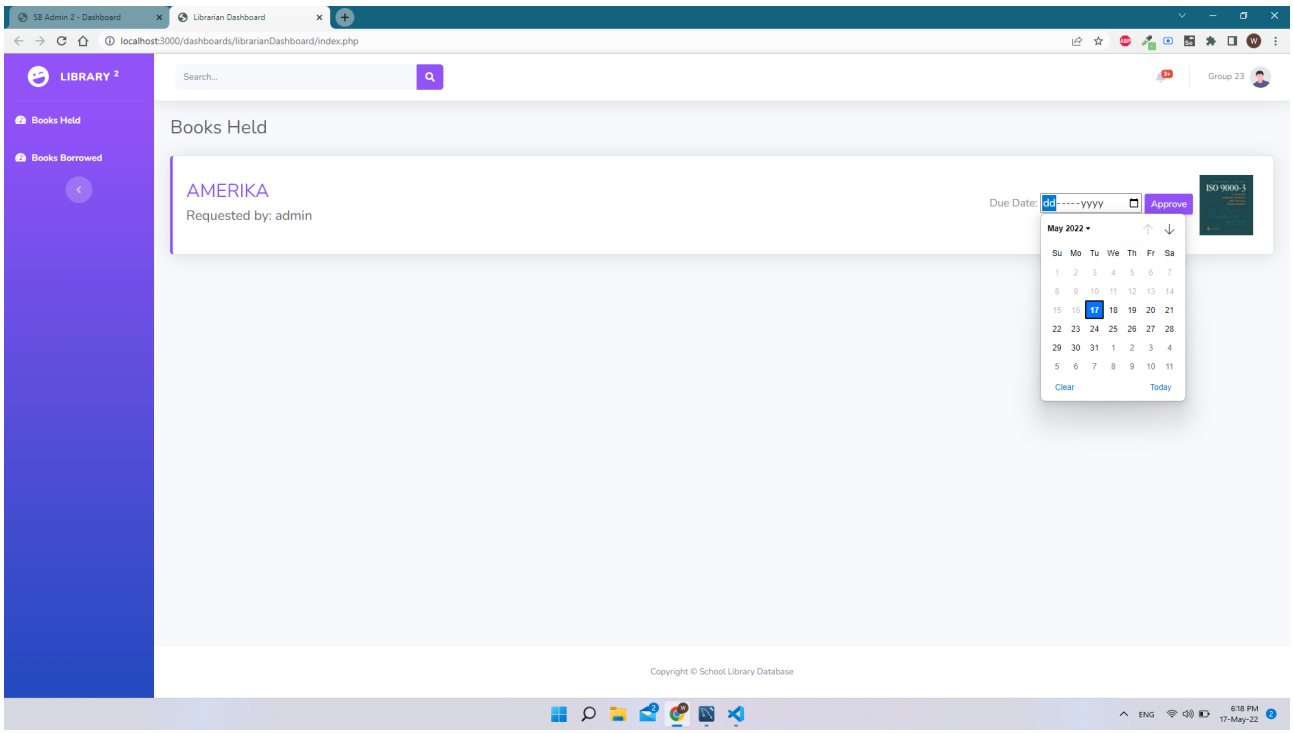

Figure 12: Book Approval Screen for Librarian with Due Date

The librarian should specify a due date for the return of a book before the approval in order to compete the approval process. This due date can be specified using the date section next to the approve button. There is no limitation for the due date and is decided by the librarian. After this process, the librarian can approve the request.

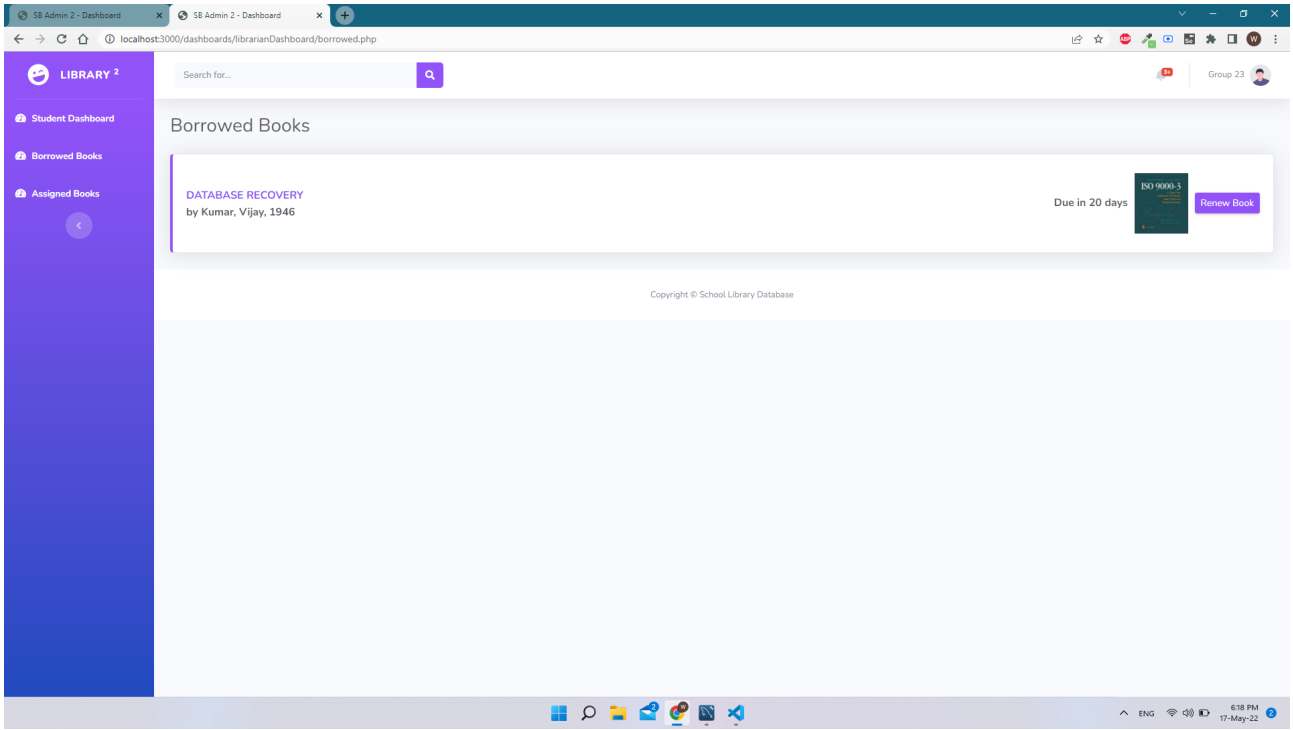

Figure 13: Borrowed Books Screen

After the approval of a book by the librarian, the user that requested the approval of the book can see the book in his/her borrowed books screen. The title, author, year, image and due date of the book is provided together with a “renew book” button which allows the user to make another approval request for the same book in order to postpone the due date.

Alers Center:

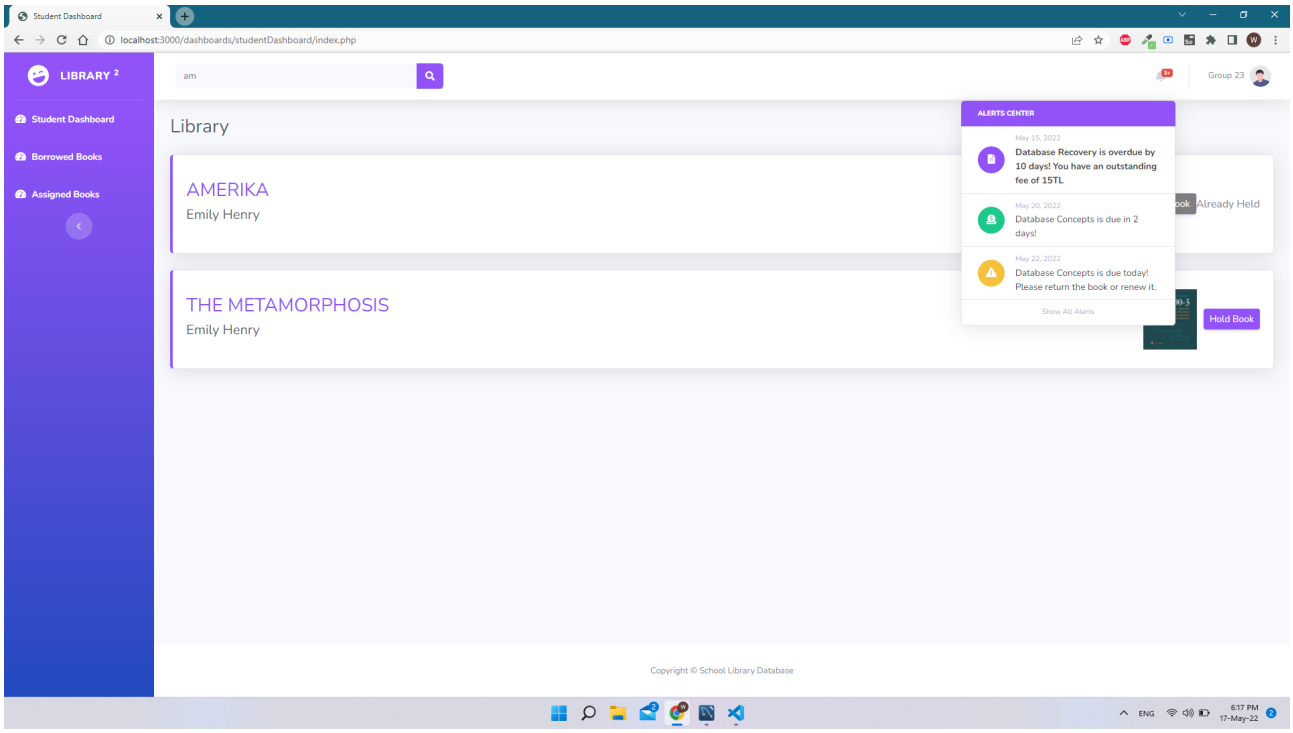

Figure 14: Alerts Center

Figure 14 represents the alerts center section of the application which enables the library management system users to display the warning messages and the notifications sent to that users in real time.

Options:

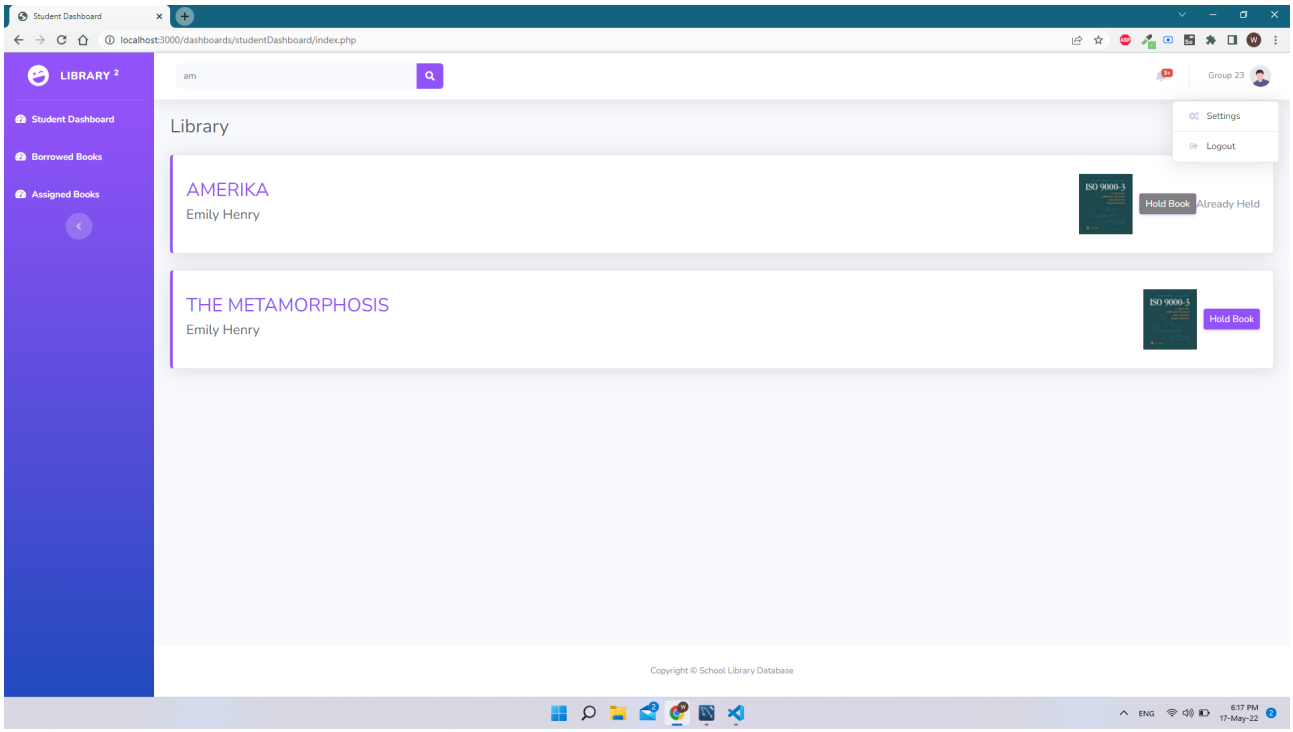

Figure 15: Options

Figure 15 represents the options dropdown menu of the application. This menu can be showed by clicking the avatar icon in the top right corner of the application. Settings and logout options are provided to the users.

Warning System:

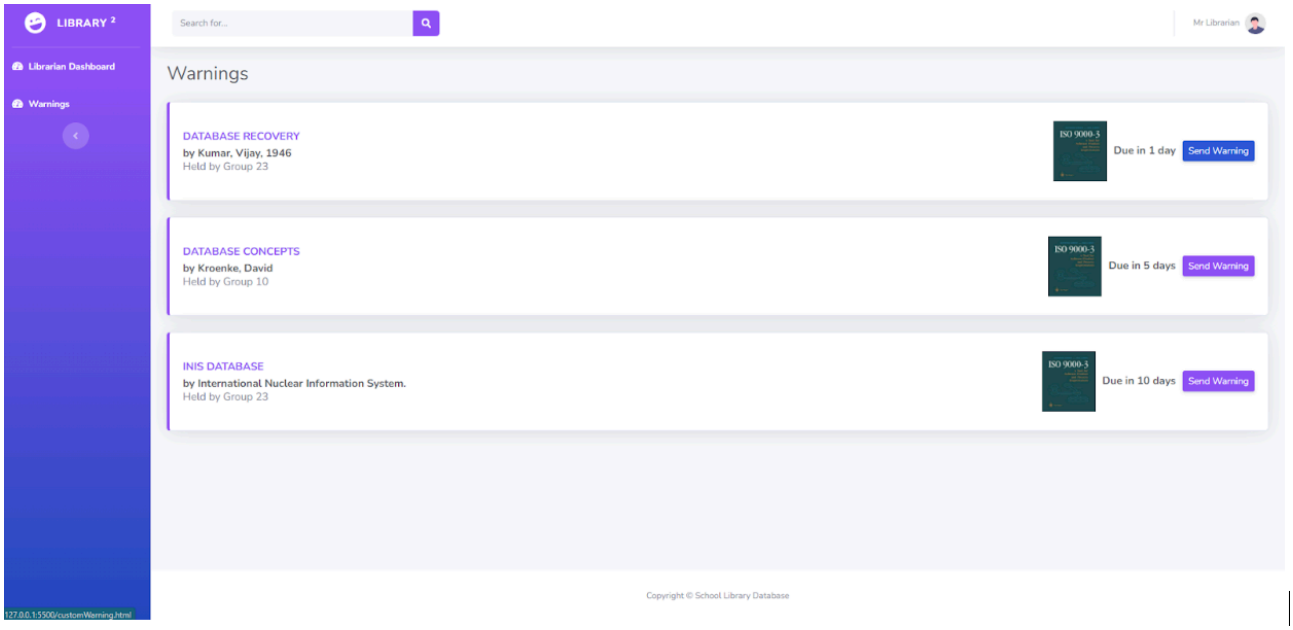

Figure 16: Send Warning

In the librarian dashboard, the librarian can see the currently borrowed books together with all the information about that book and also the user that have borrowed the book. Librarian can also see the due dates and the remaining days until the due dates of the borrowed books. In case of a need such as the exceed in a due date, the librarian can send a warning message to the relevant borrow action which means the user that have borrowed the book will be notified. Librarian can press the "Send Warning" button in order to warn the student about the corresponding borrow action.

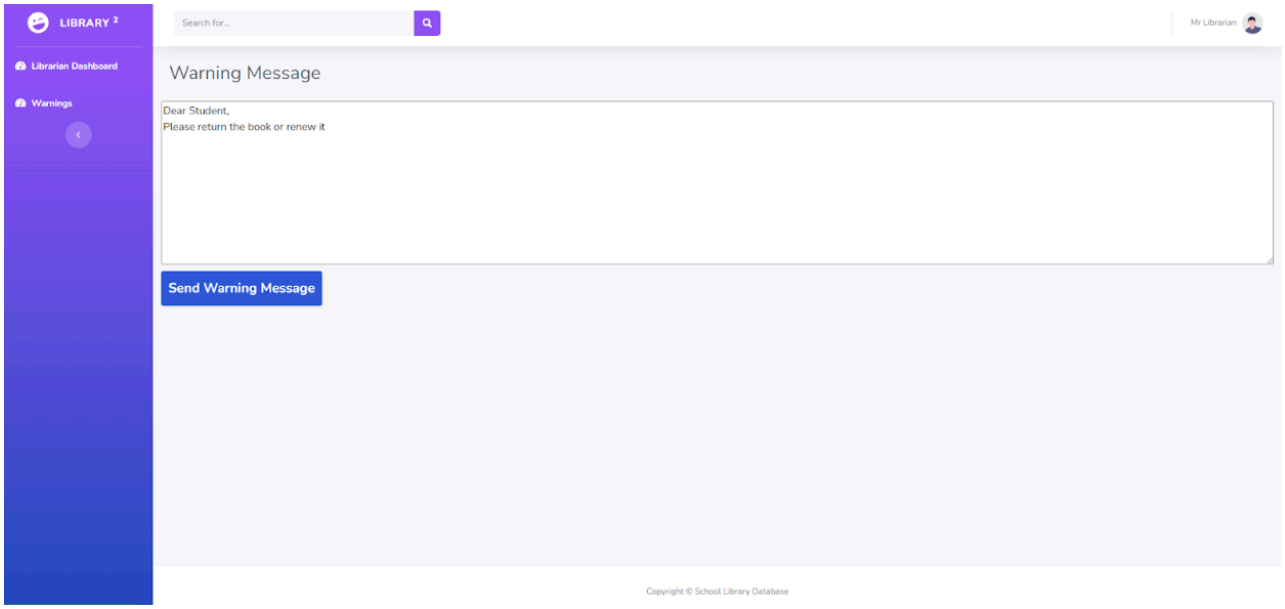

Figure 17: Warning Generation

After pressing the “Send Warning” button in *Figure 16*, Librarian is directed to the screen in *Figure 17.* In that screen the librarian can generate the warning message by filling the text area according to his/her needs. By pressing “Send Warning Message” in *Figure 17*, the librarian completes the warning notification process and the relevant warning message is sent to the relevant student according to the barrow action.

Homework Assignment:

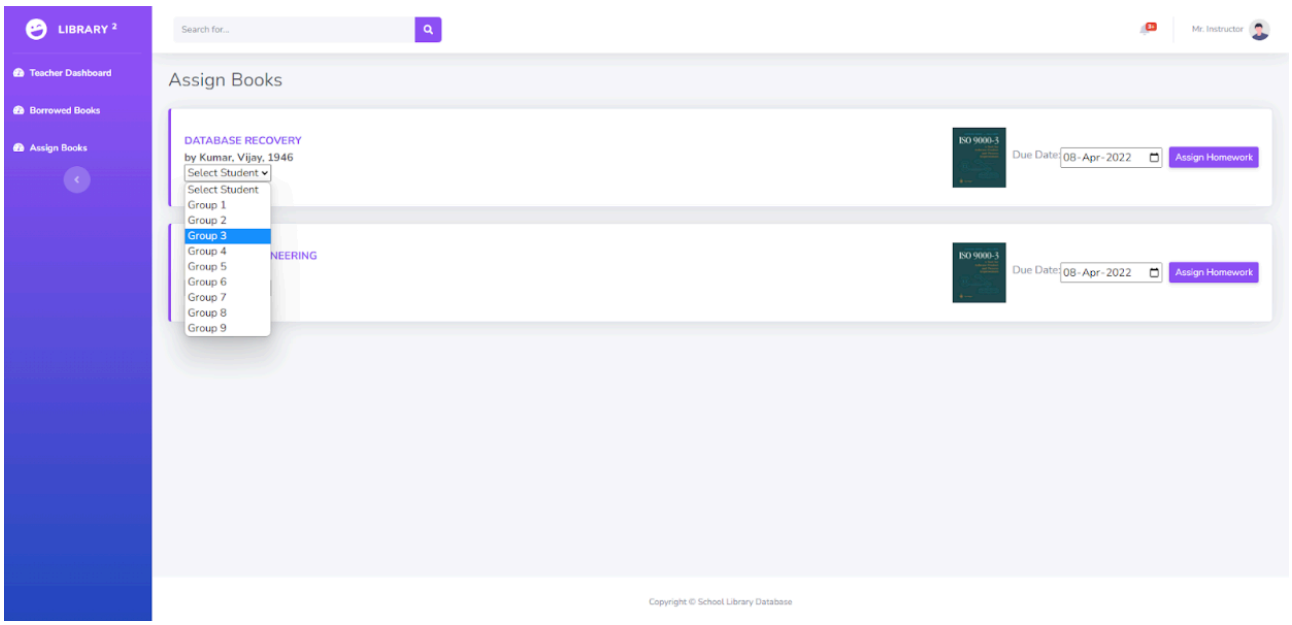

Figure 18: Homework Assignment

In the instructor’s dashboard, all the books that can be available for students’ use are listed in ascending alphabetic order according to the title of the book. As the feature of the library management system, an instructor can assign a book to a student specifying a due date. *Figure 18* represents that action. The instructor specifies the book that he/she desires to assign, then selects the student user in the dropdown menu, and finally specifies a due date for the homework assignment and finishes the action by pressing the “Assign Homework” button in the right side of the screen.

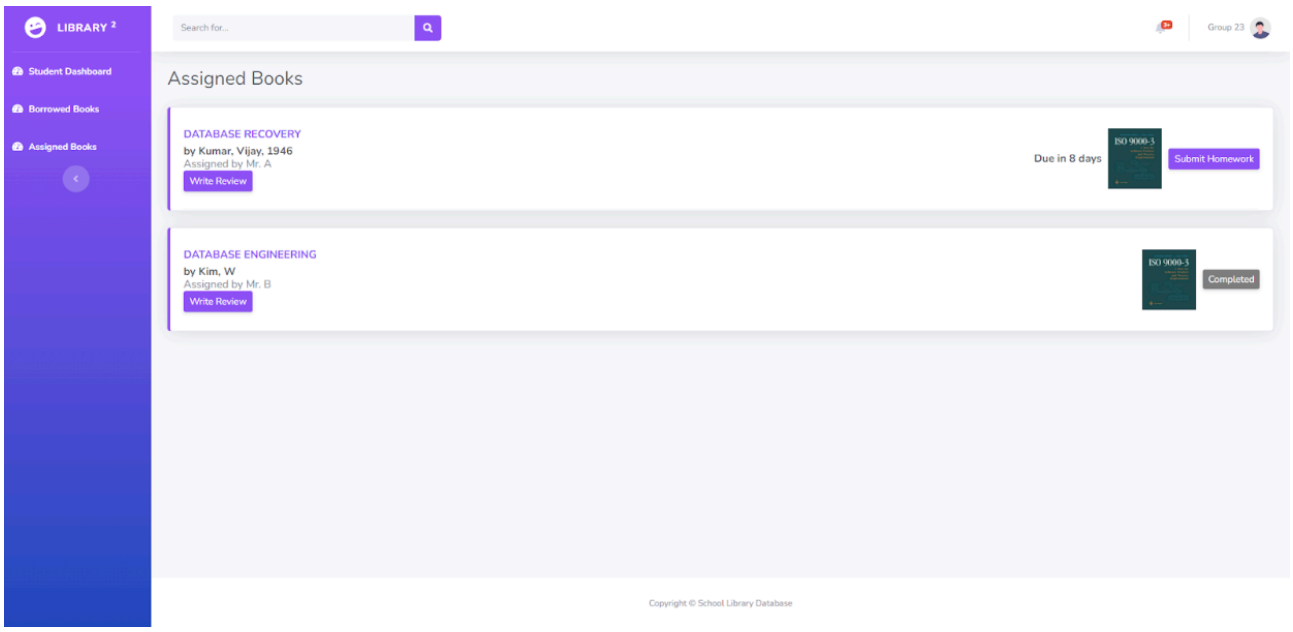

Figure 19: Homework Submission

After a homework is assigned to a student by the instructor, the student can view the assigned homework in the "Assigned Books" section on their dashboard. The relevant books that are assigned are ordered in ascending alphabetical order according to the title of the book. The author, year, image, and the instructor's name that have assigned the book is provided as information. Due dates of the homework are also provided and the due dates of the completed homework are omitted in the UI. A student can click on "Submit Homework" button in *Figure 19* in order to do the homework submission and change the status of the homework to "completed".

There is also a "Write Review" button in the assigned books section that enables the users to write their comments on the books that they are assigned with.