

DATA-236 Sec 12 - Distributed Systems for Data Engineering

HOMEWORK 4

Total Point: 20

Instructions:

- Include screenshots of the code for each component, as well as the code for app.js, along with the corresponding output. Ensure that the code and its output are displayed together, one below the other.
- Submission should be in PDF Format.
- Please name your submission file as {last_name}_HW4.pdf

Q1. React (10 Points)

Create a Book Management App where users can add, update, and delete books.

I. Home Page (1 Point)

Write the necessary code to create the 'Home.jsx' component which will display all the books. The Home component should be rendered when the user is on the root route '/'.

II. Add a New Book (2 Points)

Write the necessary code to create the 'CreateBook.jsx' component. The component should be rendered when the user is on the '/create' route.

The component should accept props (similar to demo) to add the new book and have the following:

- An input field for entering the Book Title.
- An input field for entering the Author Name.
- A submit button labeled "Add Book".

When the user clicks the Add book button:

- A new book should be added to the book list with an auto-incremented book ID.
- The user should be redirected to the home page to see the updated book list.

III. Update Book (2 Points)

Write the necessary code to create the 'UpdateBook.jsx' component. The component should be rendered when the user is on the '/update' route.

The component should accept props (similar to demo) to update the new book and have the following:

- An input field for Book Title.

- An input field for Author Name.
- A submit button labeled "Update Book".

When the user clicks the Update Book button the book within context should be updated and the user should be redirected to the home page showing updated book list.

IV. Delete the Book (2 Points)

Write the necessary code to create the 'DeleteBook.jsx' component. The component should be rendered when the user is on the '/delete' route.

The component should accept props (similar to demo) to delete a book and have the following:

- A button labeled "Delete Book".

When the user clicks the Delete Book button the book in context should be deleted and the user should be redirected to the home page showing the updated book list.

V. For all the above questions make sure to:

- Pass props for the Create, Update, and Delete components. (1 Point)
- Use hooks like useState and useEffect wherever necessary. (1 Point)
- User React-Router-Dom for routing. (1 Point)

Q2. MySQL (10 points)

In this question, you will apply the concepts learned in class to build a simple CRUD (Create, Read, Update, Delete) application using Node.js and MySQL using Sequelize

Create an appropriate book Model, *View(Optional)*, and Controller.

You are required to create a Book Management System that allows users to:

- Add a new book (POST)
- View all books (GET)
- View a book by ID (GET)
- Update book details (PUT)
- Delete a book (DELETE)

Create a MySQL database named **book_db** with a table called **books** for this task.

You are free to name your API endpoints appropriately.

Submit the Postman API response screenshots for each operation. Each completed operation will get 2 points. Also, submit a screenshot of the database and project folder structure.