

# Library Management System for Group-Based Student Borrowing

Temitope Oyemade

<https://temitopejoshua.github.io/db-personal-app/>

Department of Computer Science

Bowie State University

## 1. Executive Summary

This document outlines the design, development, and deployment of a Library Management System tailored to manage book borrowing and return processes in a classroom setting. The system addresses the needs of 78 library books, a class of students categorized into groups, and their specific borrowing preferences. It incorporates features such as borrowing status, return tracking, group membership identification, and book ratings.

## 2. Problem Definition

The challenge is to design a database system for a class with students divided into four groups (A-D), each consisting of three members. Each student has three preferred books from a longlist of 78 books. The system should manage book borrowing, return dates, and due notifications. The database must allow searching by student name to display borrowed books and corresponding due dates. Additional features include average book rating calculations and listing books by rating performance.

## 3. Requirements

- Functional Requirements:
  - Book Management: Create and manage a database of 78 books with relevant details (title, author, rating, etc).
  - Student Management: Store student information, including names, groups, and book preferences.
  - Borrowing Records: Track borrowing status, due dates, and return information.
  - Queries: Users can search by student name, group membership, and book availability.
  - Reporting: Display books with above and below-average ratings.
- Non-Functional Requirements:
  - Usability: User-friendly interface for queries.
  - Accessibility: Hosted on GitHub Pages.

## 4. Proposed Solution

We propose a Library Management System with the following features:

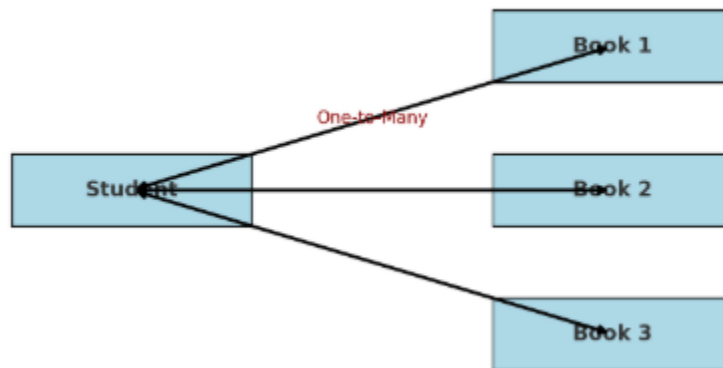
- A relational database to manage books, students, borrowing transactions, and group memberships.
- A user interface (UI) for querying borrowing details and viewing group information.
- A backend system to calculate average ratings and display books above or below the average.
- A method to dynamically update and query the database.

The system will be developed using Java for the backend, MySQL for the database, and a web interface hosted on GitHub Pages.

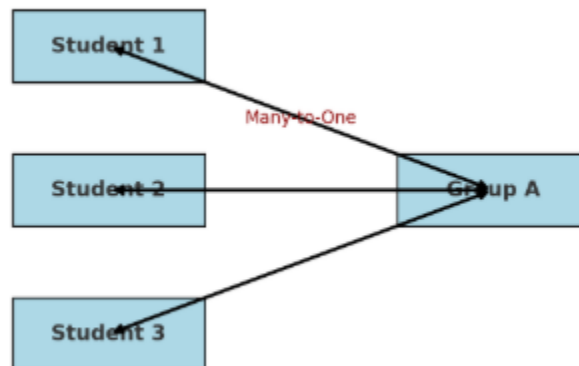
## 5. Design and Methodologies

### 5.1. Database Design

- Entities:
  - Books (BookID, Title, Author, Rating, BorrowedStatus)
  - Students (StudentID, FirstName, LastName, Group, PreferredBooks)
  - Borrowing (TransactionID, BookID, StudentID, BorrowDate, ReturnDate)
  - Groups Table: (group\_id, group\_name)
- Relationships and Entity-Relationship Diagram (ERD):
  - One-to-Many: A student can borrow multiple books.



- Many-to-One: Multiple students (3) belong to a group.



Normalization: Ensuring no data redundancy and efficient querying by dividing the database into normalized tables.

- SQL Queries:
  - Query for borrowed books and return status:

SELECT \* FROM books WHERE status = 1;

Local instance 3306 - Warning - not supported

Administration Schemas books book\_groups

1 + use library;  
2 -- Checked-in books  
3 \* SELECT \* FROM books WHERE status = 1;

100% 0 38.2

Result Grid Filter Rows Q Limit to 1000 rows Edit Export/Import

id	isbn	title	publisher_id	format	pages	published	year	status
1	978640970667	Shogun	18	paperback	112	2023-08-02 00:00:00	2023	1
2	9781028071338	Wings	3	paperback	368	2023-07-18 00:00:00	2023	1
3	9781945891181	The One and Only According to the New World	26	paperback	184	2023-03-02 00:00:00	2023	1
4	9781714502025	Time Shifter	26	hardcover	324	2022-04-27 00:00:00	2023	1
5	978194540655	The Birthday Party	27	paperback	464	2023-01-04 00:00:00	2023	1
6	9781762776827	Pyra	17	paperback	294	2023-04-08 00:00:00	2023	1
7	9781913097963	Star Born	5	paperback	300	2023-05-22 00:00:00	2023	1
8	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
9	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
10	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
11	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
12	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
13	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
14	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
15	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
16	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
17	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
18	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
19	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
20	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
21	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
22	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
23	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
24	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
25	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
26	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
27	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
28	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
29	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
30	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
31	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
32	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
33	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
34	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
35	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
36	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
37	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
38	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
39	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
40	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
41	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
42	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1
43	9781913097879	Heaven	5	paperback	178	2023-03-03 00:00:00	2023	1

books 13

Action Output

Time	Action	Response	Duration / Fetch Time
2023-07-11 10:07:55	SELECT * FROM books WHERE status = 1 LIMIT 0, 1000	13 rows returned	0.0013 sec / 0.000000

SELECT \* FROM books WHERE status = 0;

Administration

Schemas

books

book\_groups

1 + use library;

2 -- Checked-out books

3 \* SELECT \* FROM books WHERE status = 0;

100%

38.3

Result Grid

Filter Rows

Q Search

Jobs

Export/Import

id	isbn	title	publisher_id	format	pages	published	year	status
4	9781938414401	Standing Heavy	12	paperback	282	2023-05-20 00:00:00	2023	0
5	9781938414401	Standing Heavy	12	paperback	282	2023-05-20 00:00:00	2023	0
6	9781938414401	Standing Heavy	12	paperback	282	2023-05-20 00:00:00	2023	0
7	9781938414401	Standing Heavy	12	paperback	282	2023-05-20 00:00:00	2023	0
8	9781938414401	Standing Heavy	12	paperback	282	2023-05-20 00:00:00	2023	0
9	9781938414401	Standing Heavy	12	paperback	282	2023-05-20 00:00:00	2023	0
10	9781938414401	Standing Heavy	12	paperback	282	2023-05-20 00:00:00	2023	0
11	9781938414401	Standing Heavy	12	paperback	282	2023-05-20 00:00:00	2023	0
12	9781914464872	A System So Magnificent It Is Boring	29	paperback	344	2023-03-09 00:00:00	2023	0
13	9781758255007	North Building	8	paperback	272	2023-05-16 00:00:00	2023	0
14	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
15	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
16	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
17	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
18	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
19	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
20	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
21	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
22	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
23	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
24	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
25	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
26	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
27	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
28	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
29	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
30	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
31	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
32	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
33	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
34	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
35	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
36	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
37	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
38	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
39	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
40	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
41	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
42	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
43	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
44	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
45	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
46	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
47	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
48	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
49	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
50	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
51	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
52	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
53	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
54	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
55	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
56	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
57	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
58	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
59	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
60	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
61	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
62	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
63	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
64	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
65	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
66	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
67	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
68	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
69	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
70	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
71	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
72	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
73	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
74	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
75	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
76	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
77	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
78	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
79	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
80	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
81	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
82	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
83	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
84	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
85	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
86	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
87	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
88	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
89	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
90	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
91	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
92	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
93	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
94	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
95	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
96	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
97	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
98	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
99	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0
100	9781968785432	Umbra Arcana	12	paperback	143	2021-07-01 00:00:00	2022	0

books 14

100%

Action Output

1

Time

Action

Response

Duration | Fetch Time

18

2023-07-27 10:00:00

SELECT \* FROM books WHERE status = 0 LIMIT 0, 1000

25 rows(s) returned

0.0012 sec / 0.000000

- Query for group affiliations and preferred books.

```
CREATE TABLE `groups` (  
  `id` int NOT NULL AUTO_INCREMENT,  
  `name` varchar(45) DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `id_UNIQUE` (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci  
SELECT count(sg.student_id) as no_of_students_in_group, g.name FROM student_group sg,  
library.groups g  
WHERE sg.group_id = g.id GROUP BY g.name;
```

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' pane displays a tree view of the database structure, including tables like 'book\_groups', 'book\_rating', 'books', 'groups', 'student\_group', and 'students'. The 'groups' table is selected. The main editor window shows a SQL query that counts the number of students in each group. The 'Result Grid' at the bottom displays the query results as a table with two columns: 'no\_of\_students\_in\_group' and 'name'. The results show four groups: GROUP A (7 students), GROUP B (8 students), GROUP C (6 students), and GROUP D (6 students). The 'Action Output' pane at the bottom right shows the execution details, including the duration and fetch time for each row.

no_of_students_in_group	name
7	GROUP A
8	GROUP B
6	GROUP C
6	GROUP D

SELECT book\_id,b.isbn, b.title,g.name as group\_name FROM book\_groups bg, books b,  
library.groups g  
WHERE bg.book\_id = b.id AND bg.group\_id = g.id;

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'Schemas' tree with the 'library' schema selected. The main editor shows a SQL query: `use library;` followed by a comment `-- Your code should show which books for each group;` and the query: `SELECT book_id,b.isbn, b.title,g.name as group_name FROM book_groups bg, books b, library.groups g WHERE bg.book_id = b.id AND bg.group_id = g.id;`. The 'Result Grid' at the bottom displays the query results with columns: book\_id, isbn, title, and group\_name. The results are filtered to show only books belonging to 'GROUP A'.

book_id	isbn	title	group_name
81	9780261064722	The Faculty of Dreams	GROUP A
85	9781315895784	The Years	GROUP A
88	9781910701664	The 7th Function of Lan...	GROUP A
73	9781825486854	The Skunk Skyline	GROUP A
77	9781899122286	Do! My Love	GROUP A
15	9781252711238	Whales	GROUP A
16	9781839164218	In Mother Dead	GROUP A
110	9781782278627	Pyne	GROUP A
114	9781813581816	Parasite	GROUP A
118	9781839356432	China Knows	GROUP A
23	9781913067791	A New Name: Septology	GROUP A
26	9781318277182	Cursed Bunny	GROUP A
31	9781319410019	An Inventory of Lies	GROUP A
34	9781313591712	Minor Detail	GROUP A
38	9781904871401	The Pear Field	GROUP A
42	9781318174454	The Eighth Life	GROUP A
98	9781529403827	Typ	GROUP A
93	9781747305453	Mac and the Problem	GROUP A
54	9781912240166	Celestial Bodies	GROUP A
58	9780300224513	Love in the New Millenn...	GROUP A
82	9781788160619	The Pine Islands	GROUP A
68	9781786016609	Frankenstein in Baghdad	GROUP A
70	9781910701880	The Dinner Guest	GROUP A
74	978025572067	The White Book	GROUP A
78	9781915884432	Flights	GROUP A
3	9781842861131	The Gospel According to...	GROUP C
7	9781529427630	Jani Hendrix Live in Liv...	GROUP C

SELECT book\_id,b.isbn, b.title,g.name as group\_name FROM book\_groups bg, books b, library.groups g  
WHERE bg.book\_id = b.id AND bg.group\_id = g.id AND g.name = 'GROUP A';

Local Instance SQLM - Warning - not supported

Administration Schemas book book\_groups

SQLM

Filter objects

- db
- db\_class
- db\_class\_hw2
- library
- Tables
  - book\_groups
  - book\_rating
  - books
  - groups
  - student\_group
  - students
- Views
- Stored Procedures
- Functions
- sys

1 use library;

2

3 -- You code should later on add user input so that if user input Group A it should produce books belong to A, or B, or C or D

4

5 SELECT book\_id, b.isbn, b.title, g.name as group\_name FROM book\_groups bg, books b, library\_groups g

6 WHERE bg.book\_id = b.id AND bg.group\_id = g.id AND g.name = 'GROUP A';

7

8

9

100% 1/7

Result Grid Filter Rows Export

book_id	isbn	title	group_name
1	070802770007	Booker	GROUP A
2	070817002000	Time Machine	GROUP A
3	07081802770008	White Man's Burden	GROUP A
4	07081700200007	North Atlantic	GROUP A
5	07081700200008	Angry Women, Angry	GROUP A
6	07081700200009	Phenomena	GROUP A
7	07081700200010	The Book of Job	GROUP A
8	07081700200011	Whitehead	GROUP A
9	07081700200012	In Memory of Memory	GROUP A
10	07081700200013	The Endgame	GROUP A
11	070807043364	The Descent of Evening	GROUP A
12	07081700200014	The Other Name, David	GROUP A
13	07081700200015	Julia Lynn	GROUP A
14	07081700200016	Al Gosh	GROUP A
15	07081700200017	James for the Remnant	GROUP A
16	0708067054793	The Faculty of Dreams	GROUP A
17	07081700200018	The Name	GROUP A
18	07081700200019	The Two Functions of Love	GROUP A
19	07081700200020	The Book of Job	GROUP A
20	07081700200021	One, My Love	GROUP A

Result 00

Read Only

Local Instance SQLM - Warning - not supported

Administration Schemas book book\_groups

SQLM

Filter objects

- db
- db\_class
- db\_class\_hw2
- library
- Tables
  - book\_groups
  - book\_rating
  - books
  - groups
  - student\_group
  - students
- Views
- Stored Procedures
- Functions
- sys

1 use library;

2

3 -- You code should later on add user input so that if user input Group A it should produce books belong to A, or B, or C or D

4

5 SELECT book\_id, b.isbn, b.title, g.name as group\_name FROM book\_groups bg, books b, library\_groups g

6 WHERE bg.book\_id = b.id AND bg.group\_id = g.id AND g.name = 'GROUP B';

7

8

9

100% 71/4

Result Grid Filter Rows Export

book_id	isbn	title	group_name
3	0708067054793	Whale	GROUP B
4	070808704718	In Mother David	GROUP B
5	07081700200022	Pige	GROUP B
6	07081700200023	Phenomena	GROUP B
7	07081700200024	Phenomena	GROUP B
8	07081700200025	A New Name, Separately	GROUP B
9	07081700200026	Curved Runny	GROUP B
10	07081700200027	An Inventory of Losses	GROUP B
11	07081700200028	Never Lined	GROUP B
12	07081700200029	The Last Post	GROUP B
13	07081700200030	The Right Life	GROUP B
14	07081700200031	Ty	GROUP B
15	07081700200032	Mac and His Problem	GROUP B
16	07081700200033	Conductor Book	GROUP B
17	07081700200034	Love is the New Millennium	GROUP B
18	07081700200035	The First Island	GROUP B
19	07081700200036	Phenomena in the World	GROUP B
20	07081700200037	The Outer World	GROUP B
21	07081700200038	The White Book	GROUP B
22	07081700200039	Agnes	GROUP B

Result 01

Read Only

Local Instance 3306 - Warning - not supported

use library;

-- You code should later on add user input so that if user input Group A it should produce books belong to A, or B, or C or D.

```
SELECT book_id, b.isbn, b.title, g.name as group_name FROM book_groups bg, books b, library_groups g
WHERE bg.book_id = b.id AND bg.group_id = g.id AND g.name = 'GROUP C';
```

book_id	isbn	title	group_name
1	9780438611811	The Grapes According to the New World	GROUP C
7	9780102942782	Jane Hamilton Live it Live	GROUP C
11	9780153076867	Old Fern	GROUP C
13	9780100848257	Heaven	GROUP C
18	9780821581980	The Book of Esther	GROUP C
25	9780699636502	After the Sun	GROUP C
27	9780900636506	The Way of the Pilgrim	GROUP C
31	9780783275682	At Night All Blood is Black	GROUP C
36	978012087139	Summer Brother	GROUP C
38	9780602079511	The Perfect Nine	GROUP C
43	9780444444411	The Enlightenment of The Greeneyeye	GROUP C
44	9780444444411	Paras on the Top of My Tongue	GROUP C
47	978032079222	Hell Log	GROUP C
58	978019083710	Drive Your Paw Over the Bones of the	GROUP C
59	9780120207650	NEARLY 27	GROUP C
20	978011908329	The Harlequin	GROUP C
87	978044027020	Go, West, Go	GROUP C
71	9780841420121	The Flying Machine	GROUP C
74	9780786180124	The World Goes On	GROUP C

Local Instance 3306 - Warning - not supported

use library;

-- You code should later on add user input so that if user input Group A it should produce books belong to A, or B, or C or D.

```
SELECT book_id, b.isbn, b.title, g.name as group_name FROM book_groups bg, books b, library_groups g
WHERE bg.book_id = b.id AND bg.group_id = g.id AND g.name = 'GROUP D';
```

book_id	isbn	title	group_name
1	9780438611811	Standing Heavy	GROUP D
8	978034526255	The Shipyard Policy	GROUP D
12	9780191458492	A System On Mayflower it is landing	GROUP D
16	9780191284866	Love in the Big City	GROUP D
28	9780178233335	More Than I Love My Life	GROUP D
29	9780191284811	Bank of Soul	GROUP D
35	9780178233318	When We Learn to Understand the World	GROUP D
37	9780302047430	I Live in the Shark	GROUP D
39	9780178278149	The Dangers of Smoking in Bed	GROUP D
41	9780191458484	The Adventures of Chick & Bob	GROUP D
44	9780178760070	The Morning Police	GROUP D
46	9780191284797	The Purple Mirror	GROUP D
50	9780178616238	Sanctuary	GROUP D
56	9780344578237	Four Brothers	GROUP D
60	9780191134488	The Book of Moral Stories	GROUP D
64	9780807060580	The Shape of the Moon	GROUP D
86	9780178233337	Like a Flying Shadow	GROUP D
72	9780807060584	The Invisible	GROUP D
76	9780807065482	Vampire Sublime 1	GROUP D

- Query for book ratings above or below average.

Books rated higher than the median rate, median rate = 3

SELECT b.id, b.isbn, b.title, rating FROM book\_rating br, books b WHERE br.book\_id = b.id  
AND br.rating > 3;



1 use Library;

2

3 -- Books rated higher than the median rate, median rate = 3

4 SELECT book\_id, b.isbn, b.title, rating FROM book\_rating br, books b WHERE br.book\_id = b.id AND br.rating > 3

5 ORDER BY br.rating ASC;

book_id	isbn	title	rating
29	9781620975251	The Perfect Nine	4
3	9781642861161	The Gospel According to the New World	4
6	9781930174318	To Mother, I Said	4
7	9781529427820	Joni Hendrix Live in Life	4
8	9781544493655	The Birthday Party	4
9	9781854270258	When We Were Dreaming	4
10	9781782278627	Pyre	4
76	9780887055422	Vernon Scholtes 1	4
74	9780525573667	The White Book	4
14	9781911891875	Paradise	4
18	9781990368432	Elena Knows	4
19	9781982104786	The Book of Mother	4
20	9781781332835	More Than I Love My Life	4
23	9781625448554	The Silver Spade	4
24	9781911264811	Tomb of Sand	4
72	9780887056504	The Impostor	4
21	9780817424297	The Winding Mountain	4
26	9781529450793	An Inventory of Losses	4
21	9781782275862	At Night All Blood Is Black	4
30	9780300247435	I Live in the Storm	4
23	9781911891830	In Memory of Memory	4
34	9781911891772	Minor Detail	4
25	9781812867138	Sunstar Brother	4
27	9781999498865	The Synagogue	4
28	9781629425801	The Four Fads	4
77	9781999722764	Da, My Love	4
41	178057134884	The Discomfort of Evening	4
42	9781918174854	The Dying Life	4
44	9781787300750	The Memory Palace	4

Result 19

Action Output

Time Action Response Duration / Fetch Time

23 20:11:54 SELECT book\_id, b.isbn, b.title, rating FROM book\_rating br, books b WHERE br.book\_id = b.id AND br.rating > 3 30 row(s) returned 0.00061 sec / 0.000...

Books rated lower than the median rate, median rate = 3 SELECT b.id,b.isbn, b.title, rating FROM book\_rating br, books b WHERE br.book\_id = b.id AND br.rating < 3;

1 use Library;

2

3 -- Books rated lower than the median rate, median rate = 3

4 SELECT book\_id, b.isbn, b.title, rating FROM book\_rating br, books b WHERE br.book\_id = b.id AND br.rating < 3

5 ORDER BY br.rating ASC;

book_id	isbn	title	rating
76	978180956432	Right	2

Result 25

Action Output

Time Action Response Duration / Fetch Time

24 20:12:55 SELECT book\_id, b.isbn, b.title, rating FROM book\_rating br, books b WHERE br.book\_id = b.id AND br.rating < 3 1 row(s) returned 0.00060 sec / 0.000...

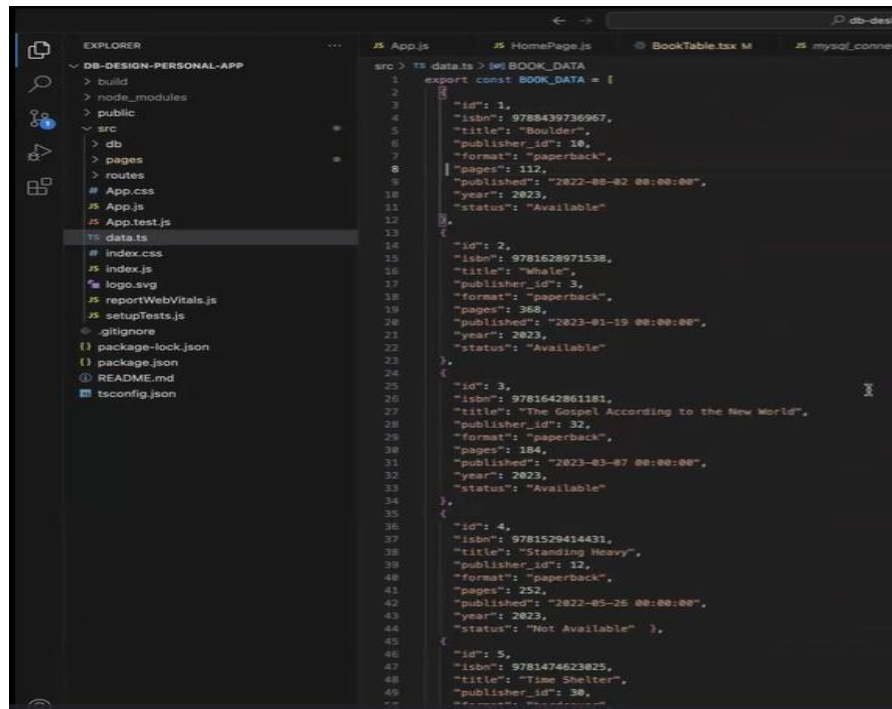
- Sequelize models: These are models to import the codes from MySQL database
  - Student data Model:



```
db-design-personal-app
src > pages > StudentsTable.tsx > ...
1 import * as React from 'react';
2 import Paper from '@mui/material/Paper';
3 import Table from '@mui/material/Table';
4 import TableBody from '@mui/material/TableBody';
5 import TableCell from '@mui/material/TableCell';
6 import TableContainer from '@mui/material/TableContainer';
7 import TableHead from '@mui/material/TableHead';
8 import TablePagination from '@mui/material/TablePagination';
9 import TableRow from '@mui/material/TableRow';
10 import TextField from '@mui/material/TextField';
11 import {STUDENT_DATA} from '../data';
12
13 interface Column {
14   id: string;
15   label: string;
16   minWidth?: number;
17   align?: 'right';
18   format?: (value: any) => string;
19 }
20
21 const columns: readonly Column[] = [
22   { id: 'id', label: 'ID', minWidth: 50 },
23   { id: 'first_name', label: 'First Name', minWidth: 150 },
24   { id: 'last_name', label: 'Last Name', minWidth: 200 },
25 ];
26
27 export default function StudentsDataTable() {
28   const [rows, setRows] = React.useState<any[]>([]);
29   const [filteredRows, setFilteredRows] = React.useState<any[]>([]);
30   const [search, setSearch] = React.useState('');
31   const [page, setPage] = React.useState(0);
32   const [rowsPerPage, setRowsPerPage] = React.useState(10);
33
34   React.useEffect(() => {
35     const loadCSV = async () => {
36       setRows(STUDENT_DATA);
37       setFilteredRows(STUDENT_DATA);
38     };
39     loadCSV();
40   }, []);
41
42   const handleChangePage = (event: unknown, newPage: number) => {
43     setPage(newPage);
44   };
45
46   const handleChangeRowsPerPage = (event: React.ChangeEvent<HTMLInputElement>) => {
47     setRowsPerPage(event.target.value);
48     setPage(0);
49   };
50 }
```

```
db-design-personal-app
src > pages > StudentsTable.tsx > ...
27 export default function StudentsDataTable() {
58
59   const handleSearch = (event: React.ChangeEvent<HTMLInputElement>) => {
60     const value = event.target.value.toLowerCase();
61     setSearch(value);
62     setFilteredRows(rows.filter(row =>
63       row.first_name.toLowerCase().includes(value) ||
64       row.last_name.toLowerCase().includes(value)
65     ));
66   };
67
68   return (
69     <div>
70       <h2>Student Data</h2>
71       <TextField
72         label="Search by first/last name"
73         variant="outlined"
74         value={search}
75         onChange={handleSearch}
76         sx={{ marginBottom: 2 }}
77       />
78       <Paper sx={{ width: '100%', overflow: 'hidden' }}>
79         <TableContainer sx={{ maxHeight: 440 }}>
80           <Table stickyHeader aria-label="sticky table">
81             <TableHead>
82               <TableRow>
83                 {columns.map((column) => (
84                   <TableCell
85                     key={column.id}
86                     align={column.align}
87                     style={{ minWidth: column.minWidth }}
88                   >
89                     {column.label}
90                   </TableCell>
91                 ))}
92             </TableHead>
93             <TableBody>
94               {filteredRows
95                 .slice(page * rowsPerPage, page * rowsPerPage + rowsPerPage)
96                 .map((row, index) => (
97                   <TableRow hover role="checkbox" tabIndex={-1} key={index}>
98                     {columns.map((column) => (
99                       const value = row[column.id];
100                       return (
101                         <TableCell key={column.id} align={column.align}>
102                           {column.format ? column.format(value) : value}
103                         </TableCell>
104                       );
105                     ))}
106                   </TableRow>
107                 ))}
108             </TableBody>
109           </Table>
110         </TableContainer>
111       </Paper>
112     </div>
113   );
114 }
```

- Book data Model:



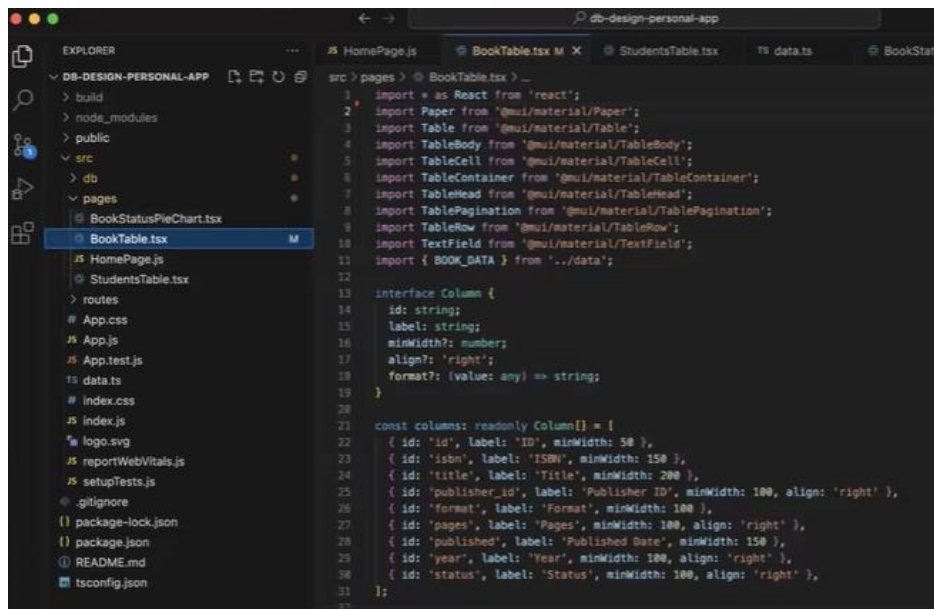
The screenshot shows a VS Code editor with the Explorer sidebar on the left displaying the project structure. The main editor window shows the `App.js` file with the following code:

```

1  export const BOOK_DATA = [
2    {
3      "id": 1,
4      "isbn": 9788439736967,
5      "title": "Boulder",
6      "publisher_id": 10,
7      "format": "paperback",
8      "pages": 112,
9      "published": "2022-08-02 00:00:00",
10     "year": 2023,
11     "status": "Available"
12   },
13   {
14     "id": 2,
15     "isbn": 9781628971538,
16     "title": "Whale",
17     "publisher_id": 3,
18     "format": "paperback",
19     "pages": 368,
20     "published": "2023-01-19 00:00:00",
21     "year": 2023,
22     "status": "Available"
23   },
24   {
25     "id": 3,
26     "isbn": 9781642861181,
27     "title": "The Gospel According to the New World",
28     "publisher_id": 32,
29     "format": "paperback",
30     "pages": 184,
31     "published": "2023-03-07 00:00:00",
32     "year": 2023,
33     "status": "Available"
34   },
35   {
36     "id": 4,
37     "isbn": 9781529414431,
38     "title": "Standing Heavy",
39     "publisher_id": 12,
40     "format": "paperback",
41     "pages": 252,
42     "published": "2022-05-26 00:00:00",
43     "year": 2023,
44     "status": "Not Available"
45   },
46   {
47     "id": 5,
48     "isbn": 9781474623025,
49     "title": "Time Shelter",
50     "publisher_id": 30,
51     "format": "paperback",
52     "pages": 112,
53     "published": "2022-08-02 00:00:00",
54     "year": 2023,
55     "status": "Available"
56   }
57 ]

```

- Book Table Model:



The screenshot shows a VS Code editor with the Explorer sidebar on the left displaying the project structure. The main editor window shows the `BookTable.tsx` file with the following code:

```

1  import * as React from 'react';
2  import Paper from '@mui/material/Paper';
3  import Table from '@mui/material/Table';
4  import TableBody from '@mui/material/TableBody';
5  import TableCell from '@mui/material/TableCell';
6  import TableContainer from '@mui/material/TableContainer';
7  import TableHead from '@mui/material/TableHead';
8  import TablePagination from '@mui/material/TablePagination';
9  import TableRow from '@mui/material/TableRow';
10 import TextField from '@mui/material/TextField';
11 import { BOOK_DATA } from '../data';
12
13 interface Column {
14   id: string;
15   label: string;
16   minWidth?: number;
17   align?: 'right';
18   format?: (value: any) => string;
19 }
20
21 const columns: readonly Column[] = [
22   { id: 'id', label: 'ID', minWidth: 50 },
23   { id: 'isbn', label: 'ISBN', minWidth: 150 },
24   { id: 'title', label: 'Title', minWidth: 200 },
25   { id: 'publisher_id', label: 'Publisher ID', minWidth: 100, align: 'right' },
26   { id: 'format', label: 'Format', minWidth: 100 },
27   { id: 'pages', label: 'Pages', minWidth: 100, align: 'right' },
28   { id: 'published', label: 'Published Date', minWidth: 150 },
29   { id: 'year', label: 'Year', minWidth: 100, align: 'right' },
30   { id: 'status', label: 'Status', minWidth: 100, align: 'right' },
31 ];

```

- Database Connector:

```

src > db > sequelize-project > mysql_connector.js >
1  const { Sequelize } = require('sequelize');
2
3
4  const dbConnector = new Sequelize({
5    database: 'library',
6    username: 'capricorn',
7    password: 'welcome123',
8    host: 'localhost',
9    port: 3386,
10   dialect: 'mysql',
11 })
12
13 dbConnector.authenticate().then(() => {
14   console.log('Connected to the database successfully!')
15 }).catch(error => console.log('An error occured when trying to connect to the database ' + error))
16
17 module.exports = {
18   dbConnector
19 }
20

```

- Routers: These were used to connect the front-end our backend database
  - React: The front end was built with React

```

src > index.js >
1  import React from 'react';
2  import ReactDOM from 'react-dom/client';
3  import './index.css';
4  import App from './App';
5  import reportWebVitals from './reportWebVitals';
6
7
8  const root = ReactDOM.createRoot(document.getElementById('root'));
9  root.render(
10    <React.StrictMode>
11      <App />
12    </React.StrictMode>
13  );
14
15 // If you want to start measuring performance in your app, pass a function
16 // to log results (for example: reportWebVitals(console.log))
17 // or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
18 reportWebVitals();
19

```

- HashRoute: HashRoute was imported for connectivity with GitHub

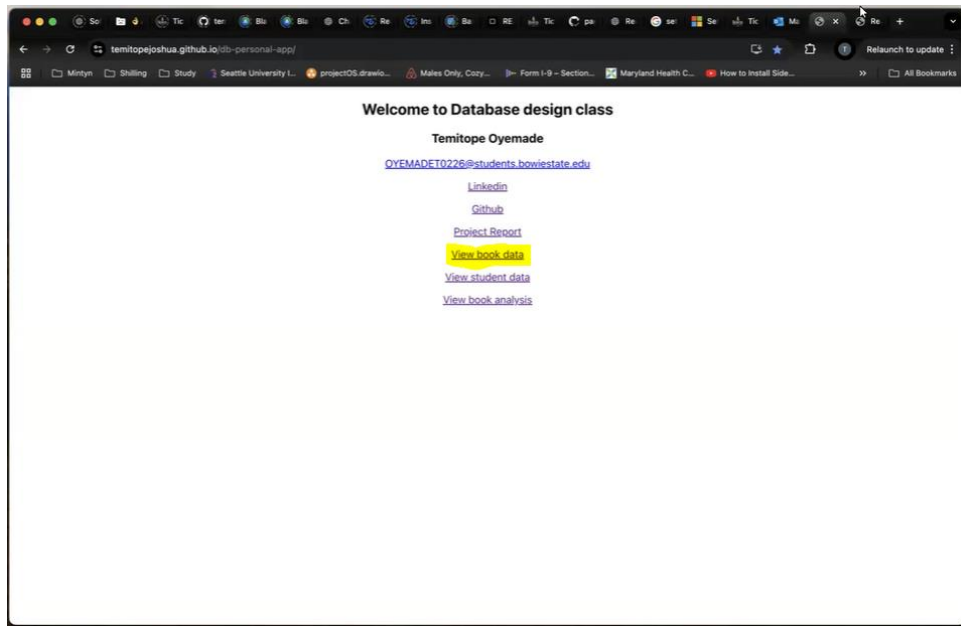
```

src > routes > NavigationRoutes.js > RootNavigation
1  import React from 'react';
2  import { Route, Routes } from 'react-router-dom';
3  import HomePage from '../pages/HomePage';
4  import StickyHeadTable from '../pages/BookTable';
5  import StudentsDataTable from '../pages/StudentsTable';
6  import BookStatusPieChart from '../pages/BookStatusPieChart';
7  import { HashRouter } from 'react-router-dom';
8
9
10
11 function RootNavigation() {
12   return (
13     <HashRouter>
14       <Routes>
15         <Route path="/" element={<HomePage/>} />
16         <Route path="/book-data" element={<StickyHeadTable/>} />
17         <Route path="/student-data" element={<StudentsDataTable/>} />
18         <Route path="/book-status-analysis" element={<BookStatusPieChart/>} />
19       </Routes>
20     </HashRouter>
21   );
22 }
23
24 export default RootNavigation;
25

```

## 5.2. Front-end Demonstration

On the website, click on “View book data” to view the book data



Webpage displaying Book Data

**Book Data**

Search by title

ID	ISBN	Title	Publisher ID	Format	Pages	Published Date	Year	Status
1	9788439736967	Boulder	10	paperback	112	2022-08-02 00:00:00	2023	Available
2	9781628971538	Whale	3	paperback	368	2023-01-19 00:00:00	2023	Available
3	9781642861181	The Gospel According to the New World	32	paperback	184	2023-03-07 00:00:00	2023	Available
4	9781529414431	Standing Heavy	12	paperback	252	2022-05-26 00:00:00	2023	Not Available
5	9781474623025	Time Shelter	30	hardcover	304	2022-04-21 00:00:00	2023	Available
6	9781839764318	Is Mother Dead	29	paperback	330	2022-09-27 00:00:00	2023	Not Available
7	9781529427820	Jimi Hendrix Live in Lviv	12	hardcover	416	2023-04-27 00:00:00	2023	Not Available

Rows per page: 10 1-10 of 78

Search by Title – for example Whale, or Paradais – as you begin to type, it begins to filter out your search

temitopejoshua.github.io/db-personal-app/#/book-data

### Book Data

Search by title

whale

ID	ISBN	Title	Publisher ID	Format	Pages	Published Date	Year	Status
2	9781628971538	Whale	3	paperback	368	2023-01-19 00:00:00	2023	Available

Rows per page: 10 1-1 of 1

temitopejoshua.github.io/db-personal-app/#/book-data

### Book Data

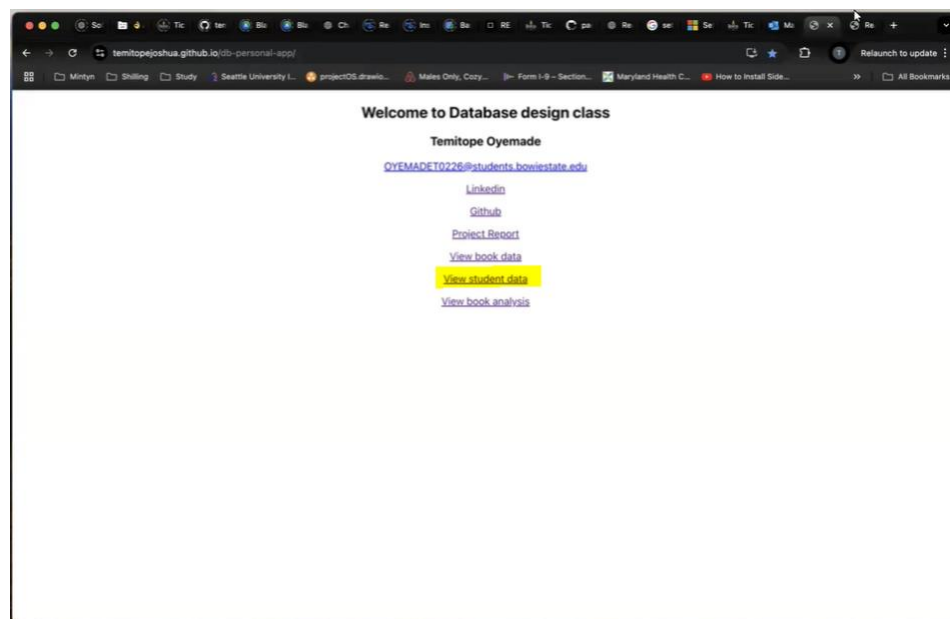
Search by title

pa

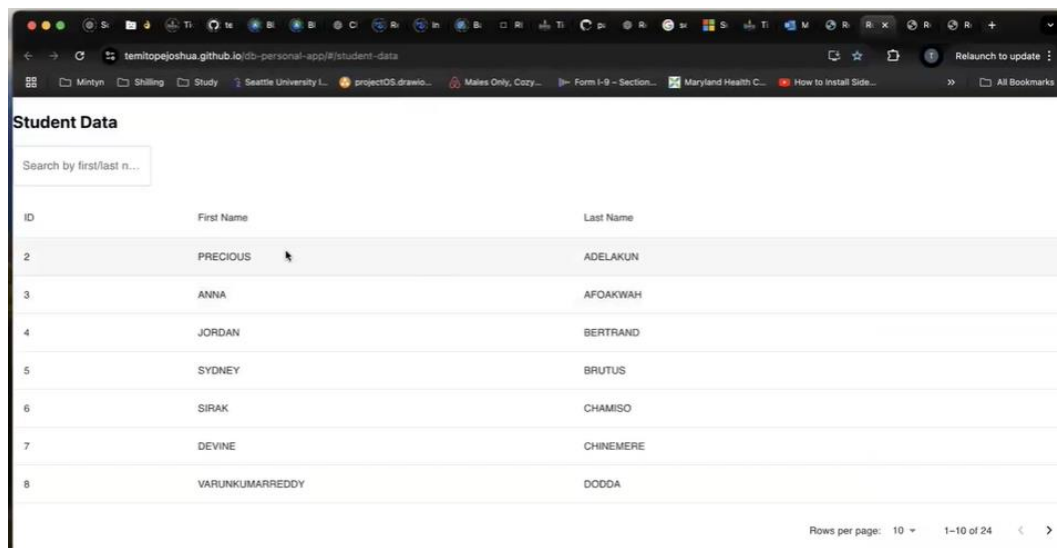
ID	ISBN	Title	Publisher ID	Format	Pages	Published Date	Year	Status
8	9781945492655	The Birthday Party	27	paperback	454	2023-01-24 00:00:00	2023	Available
14	9781913097875	Paradise	5	paperback	118	2022-03-23 00:00:00	2022	Available

Rows per page: 10 1-2 of 2

On the website, Click on “View student data” to view student data



## Webpage displaying Student Data



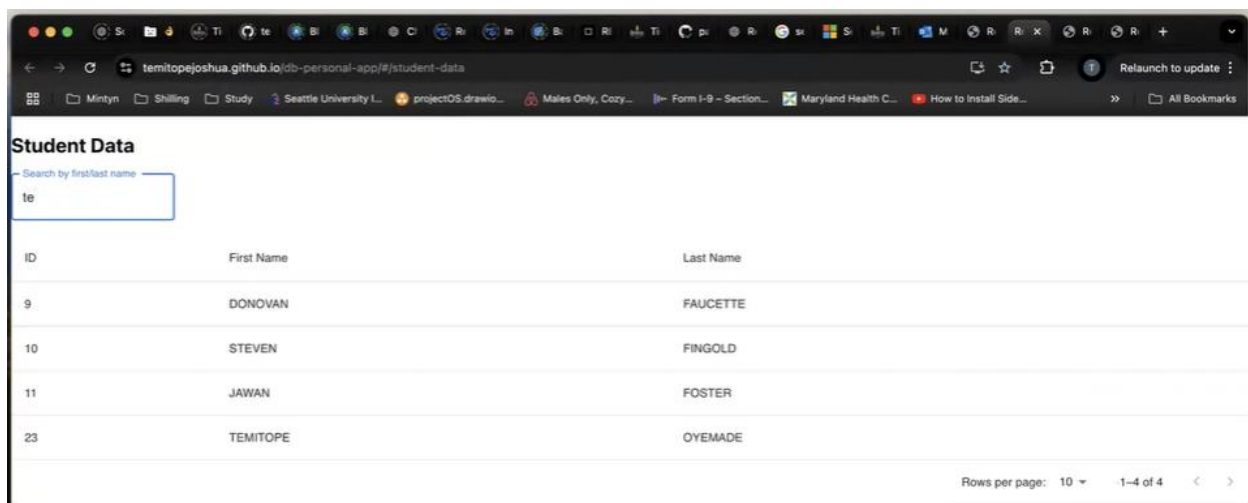
Student Data

Search by first/last n...

ID	First Name	Last Name
2	PRECIOUS	ADELAUN
3	ANNA	AFOAKWAH
4	JORDAN	BERTRAND
5	SYDNEY	BRUTUS
6	SIRAK	CHAMISO
7	DEVINE	CHINEMERE
8	VARUNKUMARREDDY	DODDA

Rows per page: 10 1-10 of 24

Users can search student data by first or last name. For example as you begin to type “te” for Temitope all the name that have “te” are displayed or “ja” for Jared, all the name with “ja” are displayed.



Student Data

Search by first/last name

te

ID	First Name	Last Name
9	DONOVAN	FAUCETTE
10	STEVEN	FINGOLD
11	JAWAN	FOSTER
23	TEMITOPE	OYEMADE

Rows per page: 10 1-4 of 4



Student Data

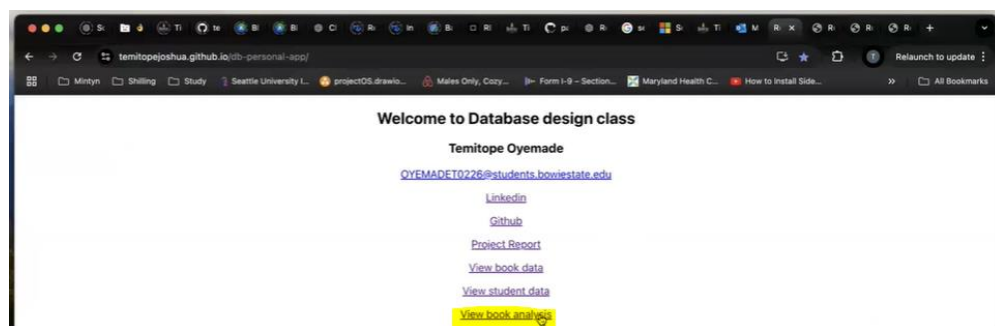
Search by first/last name

ja

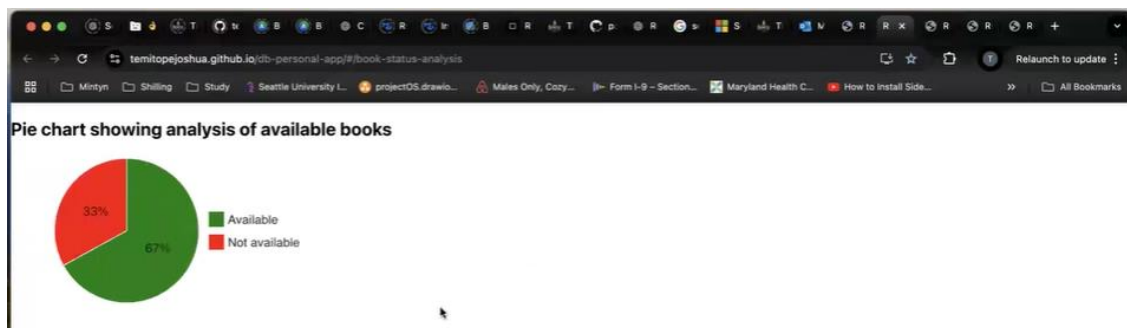
ID	First Name	Last Name
11	JAWAN	FOSTER
24	JARED	ROBINSON

Rows per page: 10 1-2 of 2

On the front end, Click on “View book analysis” to view the data analysis of book availability.

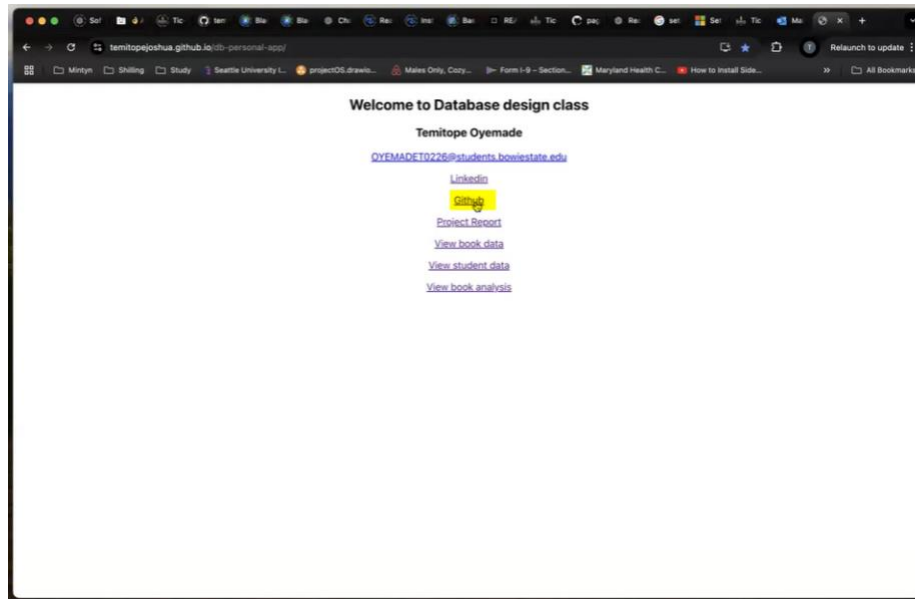


The pie chart shows a quick view of the percentage of books that are available and unavailable.

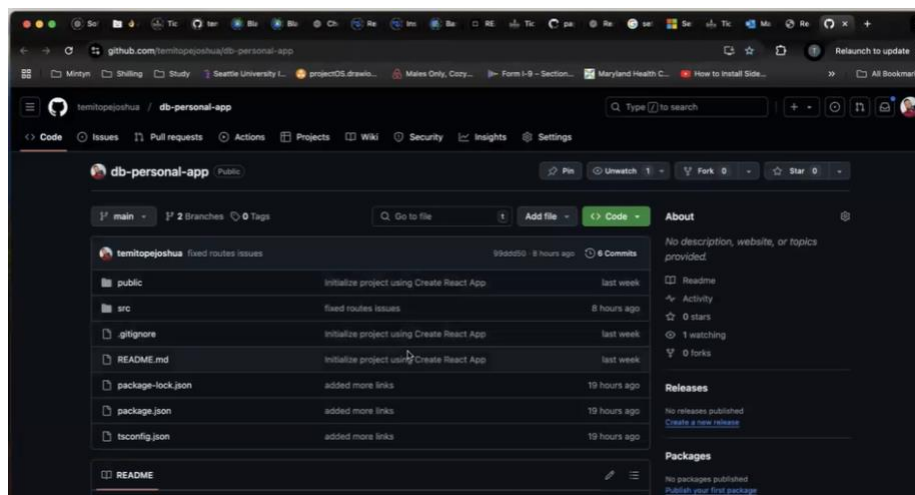


On the website, Click on “GitHub” to view the GitHub Page.





Website displaying GitHub Page with all the project packages and codes



### 5.3. Tools and Technologies

- **Programming Language:**
  - **Java:** Chosen for its platform independence, strong database connectivity, and robust features, Java was used to build the core application logic for interactive database management.
- **Backend Framework:**
  - **Node.js:** Utilized for its lightweight, event-driven architecture, Node.js efficiently handled concurrent requests and facilitated dynamic interactions between the client and the database.
- **ORM (Object-Relational Mapping):**
  - **Sequelize:** This ORM simplified database operations with MySQL, providing an abstraction layer over raw SQL queries. This reduced development time, enabled clean code, and improved maintainability.

- Database:
  - MySQL: Selected for its reliability, scalability, and support for SQL standards, MySQL efficiently managed the relational data needed for the project. Its robust querying capabilities allowed seamless handling of borrowing and returning operations.
- Web Hosting:
  - GitHub Pages: Used to host project documentation and static resources, ensuring easy public access, version control, and collaborative development.
- Frontend Framework:
  - React with HashRouter: React enabled the development of a dynamic and responsive user interface, while HashRouter provided seamless navigation with GitHub.

## 6. System Features

### 6.1. Borrowing Management

- Specify the name of the student.
- Display borrowed books, borrower details, and return dates.
- Dynamically update the borrowing and return status.

### 6.2. Rating Analysis

- Show books rated higher or lower than the average rating.
- Allow users to input new ratings to update average dynamically.

### 6.3. Group Management

- Display students in groups (A, B, C, D).
- Show books borrowed by students in each group.
- Allow querying of a student's group based on their name.

## 7. Conclusion and Recommendations

The Library Book Borrowing Management System successfully manages book inventory, student assignments, and borrowing activities while enabling meaningful queries and reports. Future improvements could include automated notifications for due dates and overdue penalties.

## Appendices

Source Code Repository Link: <https://github.com/temitopejoshua/db-personal-app>

GitHub Page Link: <https://temitopejoshua.github.io/db-personal-app/>