



1 (10 pt). You need to design a document-based database for this system. You should use MongoDB Atlas as a free cloud-based DBMS. Then, you fill your database with at least 10 users, 2 library managers, and 30 books. Each book has randomly 2-5 copies.

I have created the database and the collections according to the requirements in the MongoDB Atlas or the MongoDB Compass. The name of the database is “dbms” and the names of the collections are ‘users’, ‘books’ and ‘loans’. I have stored the database collections in the MongoDB Data folder in two types of format which are json and csv. The users database has about 10 users and 3 library managers and around 33 books where each book has randomly 2-5 copies.

Image of the books collection:

The screenshot shows the MongoDB Compass interface for the dbms.books collection. The left sidebar shows databases like admin, book_search, dbms (which is selected), and local. The main area displays the dbms.books collection with 34 documents and 1 index. It includes tabs for Documents, Aggregations, Schema, Indexes, and Validation. A search bar and filter options are available. Two documents are previewed:

```
_id: 18
title: "Seamless demand-driven strategy"
description: "Trial safe pull act. Blood deal loss after have natural.
Guy before ru."
price: 48
author: "Steven Shelton"
publisher: "Drake-Howard"
copies: Array (3)
--v: 1

_id: 5
title: "Monitored exuding superstructure"
description: "Politics sort meeting nor. Wife police animal throw respond.
You offic--"
price: 75
author: "Jeffrey Harrison"
publisher: "Kelly-Donaldson"
copies: Array (6)
--v: 2
```



Image of users collection:

MongoDB Compass - pro 2/dbms.users

dbms.users

Documents Aggregations Schema Indexes Validation

Filter Type a query: { field: 'value' }

13 DOCUMENTS 1 INDEXES

ADD DATA EXPORT DATA Explain Reset Find Options

1 - 13 of 13

Three document cards are visible:

```
_id: 10
username: "rsutton"
password: "nPf3phshu2"
full_name: "Bryan Ingram"
address: "651 Paul Trafficway, Jeffreybury, GA 78254"
phone_number: "(731)358-9483x9163"
email: "jacksonname@example.com"
role: "library_manager"
```

```
_id: 7
username: "tamaraafrazier"
password: "x%a00ZfBep"
full_name: "Richard Moyer II"
address: "3335 Wood Viaduct, Beckyhaven, AZ 28240"
phone_number: "+1-605-226-7553"
email: "susan27@example.com"
role: "user"
```

```
_id: 5
username: "burtonwilliam"
password: "168022r+FJ"
```

Image of loans collection:

MongoDB Compass - pro 2/dbms.loans

dbms.loans

Documents Aggregations Schema Indexes Validation

Filter Type a query: { field: 'value' }

58 DOCUMENTS 1 INDEXES

ADD DATA EXPORT DATA Explain Reset Find Options

21 - 40 of 58

Three document cards are visible:

```
_id: ObjectId('64bcc42f1c6c02189dd11fd')
userId: 4
bookId: 6
borrowDate: 2023-07-23T06:09:51.110+00:00
dueDate: 2023-07-30T06:09:51.110+00:00
__v: 0
```

```
_id: ObjectId('64bcdd9a9c6f92095a0b3e25')
userId: 9
bookId: 5
borrowDate: 2023-07-23T07:58:18.106+00:00
dueDate: 2023-07-30T07:58:18.106+00:00
__v: 0
```

```
_id: ObjectId('64bcdd9f9c6f92095a0b3e27')
userId: 9
bookId: 7
borrowDate: 2023-07-23T07:58:23.943+00:00
dueDate: 2023-07-30T07:58:23.943+00:00
__v: 0
```



2 (20 pt). Develop a GUI-based software application to demonstrate this online shopping system.

Database Used: MongoDB Atlas and/or MongoDB Compass.

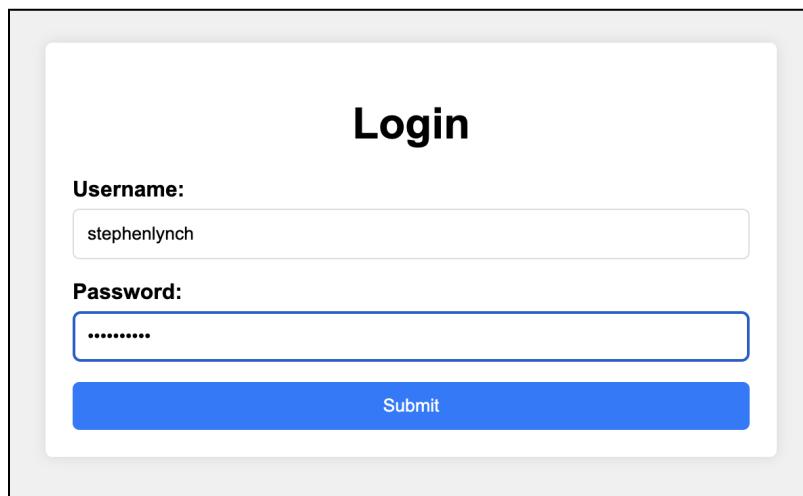
Languages or the frameworks used: node.js, express.js, HTML, CSS, and JavaScript.

To run: npm install

To start the server: node server.js

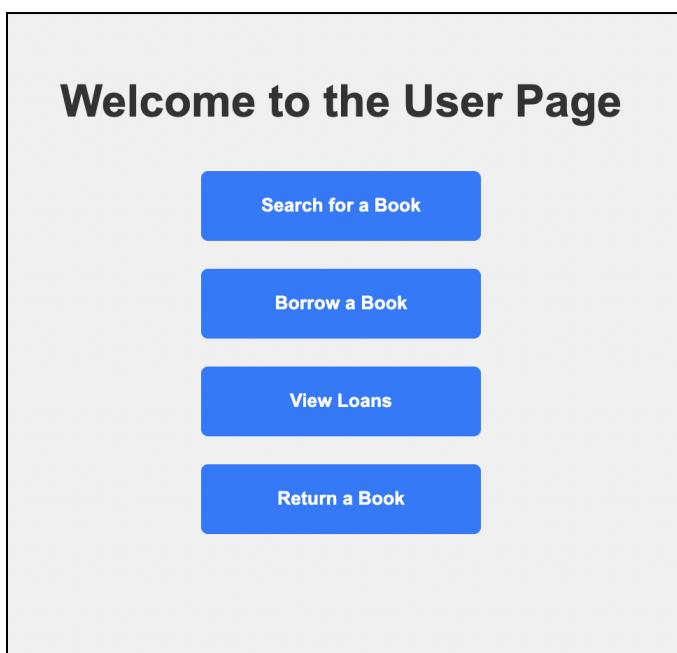
- a. (2 pt) Login to the system with his/her username and password;

<http://localhost:3000/user>



A screenshot of a login form titled "Login". It has two input fields: "Username" containing "stephenlynch" and "Password" containing ".....". A blue "Submit" button is at the bottom.

After logging in with their credentials, the user will be redirected to the appropriate page based on their role. Users will be taken to the users page, and managers will be taken to the managers page.



A screenshot of the "Welcome to the User Page". It features four blue buttons with white text: "Search for a Book", "Borrow a Book", "View Loans", and "Return a Book".



b. (3 pt) Borrow a book copy;

To borrow a book, click the borrow button.

Borrow a Book

Your ID:

Book ID:

Borrow

Book borrowed successfully

See My Loans

The user will enter their user ID and the book ID of the book they want to borrow. When the user clicks the "Borrow" button, if the message "Book borrowed successfully" appears, the borrow was successful and the user has successfully borrowed a copy of that book.

The user can also see their loans by clicking "See My Loans" button:

My Loans

Book ID: 24, Borrow Date: 2023-07-28T01:41:19.673Z, Due Date: 2023-08-04T01:41:19.673Z

Return

Here, the user can see the book they have borrowed. The copies of the books are stored in an array.



c. (3 pt) Return a book copy;

To return a book, the user needs to enter the book ID. The user can view the ID on their loan page. Each user has a separate loan page where they can view their own loans.

Return a Book

Book ID:

24

Return Book

After entering the required book ID, if the user clicks the "Return Book" button, they should see a pop-up that says "Book returned successfully". If the user sees this pop-up, the return was successful.

A screenshot of a web browser window. The address bar shows 'localhost:3000/return'. A modal dialog box is open in the center, displaying the message 'localhost:3000 says Book returned successfully' with an 'OK' button. Below the modal, the main content area shows a 'Return a Book' form with a 'Book ID:' field containing '24' and a 'Return Book' button.

To check if the book has been returned successfully, we can go back to the loans page and refresh it. If the book has been returned successfully, the loan page will not show any book borrowing records for the book ID 24.

A screenshot of a web browser window. The address bar at the top shows the URL "localhost:3000/loan.html". The main content area of the browser is titled "My Loans" in bold black text, and the rest of the page is blank white space.

d. (3 pt) Add a new book.

To add a new book, the manager must first log in. Once logged in, the manager can add a new book by providing the book's title, author, genre, and number of copies.

A screenshot of a login form. The title "Login" is centered at the top in a large, bold, black font. Below it is a "Username:" label followed by a text input field containing the value "lisa". Below that is a "Password:" label followed by a password input field containing four dots ("...."). At the bottom of the form is a blue "Submit" button.

Lisa, as a library manager, has access to the following options.

A screenshot of a dashboard for a library manager. The title "Welcome Manager!" is centered at the top in a bold, black font. Below the title are six blue rectangular buttons arranged in two rows of three. The top row contains "Add Book", "Update Book", and "Add Copy". The bottom row contains "Remove Copy", "View Loans", "Extend Loan Due Date", and "Popular Books".

The library manager can select any of the above options they wanted to do.



Adding a new book:

Add a new book

Book ID:

Title:

Description:

Price:

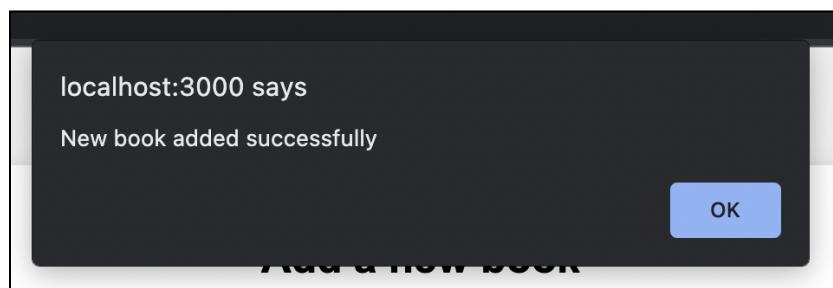
Author:

Publisher:

Copies (separated by commas):

Add Book

The library manager is adding a new book with the above details. Once the book has been added successfully and updated in the database, a pop-up will appear as shown below.



The details of the newly added book in MongoDB are shown in the image below.



dbms.books

35 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Indexes Validation

Filter Type a query: { field: 'value' } Explain Reset Find Options ▾

ADD DATA EXPORT DATA ▾ 21 - 35 of 35 ⏪ ⏴ ⏵ ⏷ ⏸ ⏹

copies: Array (5)
__v: 1

_id: 333
title: "Eras Tour _"
description: "Tour by Taylor Swift"
price: 100
author: "Taylor Swift"
publisher: "TS"
copies: Array (7)
__v: 1

_id: 5555
title: "Evermore (Taylor Swift Book)"
description: "Like its predecessor, Evermore is also a product of remote collaborati..."
price: 50
author: "Taylor Swift"
publisher: "Republic Records"
copies: Array (5)
__v: 0

e. (3 pt) Update information of a book.

Updating the information of the book with the following details:

Update Book

Book ID:
5555

Title:
Evermore

Description:
Like its predecessor, Evermore is also a product of remote collaboration and virtu

Price:
100

Author:
Taylor Swift

Publisher:
Republic Records LLC

Copies:
2

Update Book



Details of the book before updating its information:

The screenshot shows the dbms.books interface with the following details:

dbms.books (Collection name)

35 DOCUMENTS | **1 INDEXES**

Documents (selected), Aggregations, Schema, Indexes, Validation

Filter (dropdown), Type a query: { field: 'value' } (query input), Explain, Reset, Find, Options ▾

ADD DATA (button), **EXPORT DATA** (button), Page: 21 - 35 of 35, Navigation icons, List View, Grid View, Details View, Row Selection

Document 1 (highlighted):

```
_id: 333
title: "Eras Tour _"
description: "Tour by Taylor Swift"
price: 100
author: "Taylor Swift"
publisher: "TS"
copies: Array (5)
__v: 1
```

Document 2:

```
_id: 5555
title: "Evermore (Taylor Swift Book)"
description: "Like its predecessor, Evermore is also a product of remote collaborati..."
price: 50
author: "Taylor Swift"
publisher: "Republic Records"
copies: Array (5)
__v: 0
```

Details of the book after updating its information:

The screenshot shows the dbms.books interface with the following details:

dbms.books (Collection name)

35 DOCUMENTS | **1 INDEXES**

Documents (selected), Aggregations, Schema, Indexes, Validation

Filter (dropdown), Type a query: { field: 'value' } (query input), Explain, Reset, Find, Options ▾

ADD DATA (button), **EXPORT DATA** (button), Page: 21 - 35 of 35, Navigation icons, List View, Grid View, Details View, Row Selection

Document 1 (highlighted):

```
_id: 333
title: "Eras Tour _"
description: "Tour by Taylor Swift"
price: 100
author: "Taylor Swift"
publisher: "TS"
copies: Array (7)
__v: 1
```

Document 2:

```
_id: 5555
title: "Evermore"
description: "Like its predecessor, Evermore is also a product of remote collaborati..."
price: 100
author: "Taylor Swift"
publisher: "Republic Records LLC"
copies: Array (5)
__v: 1
```

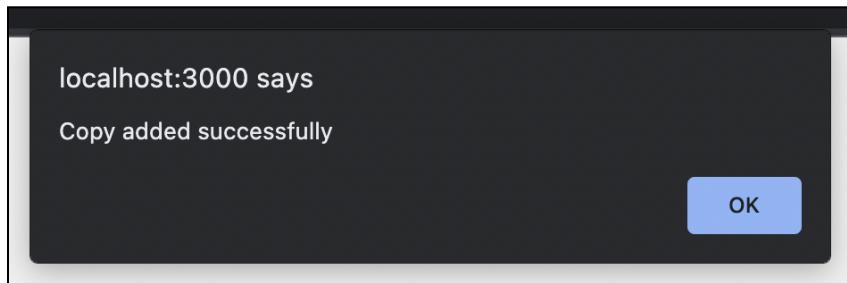


f. (2 pt) Add or remove a copy for a book.

To add a copy of a book, the manager should click the Add Copy button on the manager page.

The screenshot shows a modal window titled "Add Book Copy". It has two input fields: "Book ID:" containing "5555" and "Copy Number:" containing "9". Below the inputs is a blue "Add Copy" button.

When the copy of the book has been added successfully, a pop-up will appear that says “Copy added successfully”.



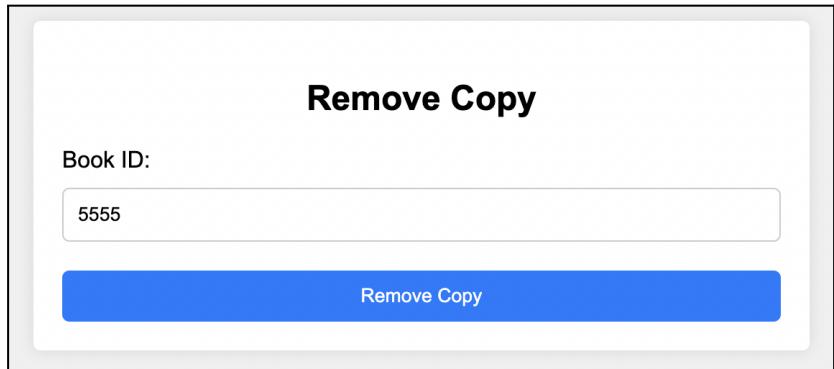
The below image shows the image of the database, where the copy of the book with id 5555, has been added successfully.

```
_id: 5555
title: "Evermore"
description: "Like its predecessor, Evermore is also a product of remote collaborati..."
price: 100
author: "Taylor Swift"
publisher: "Republic Records LLC"
copies: Array (6)
__v: 0
```

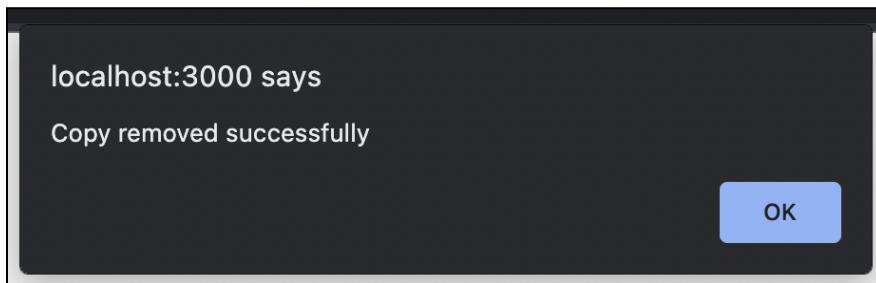
The number of copies has been updated from Array (5) to Array (6). The copies of the books are stored in an array.



To remove a copy of a book, the manager should click the Remove Copy button on the manager page.



You can mention the book id or the copy number of that book, to remove the copy of the book. When the copy of the book has been removed successfully, a pop-up will appear that says "Copy removed successfully".



The below image shows the image of the database, where the copy of the book with id 5555, has been removed successfully.

```
_id: 5555
title: "Evermore"
description: "Like its predecessor, Evermore is also a product of remote collaborati..."
price: 100
author: "Taylor Swift"
publisher: "Republic Records LLC"
copies: Array (5)
--v: 1
```

The number of copies has been updated from Array (6) to Array (5). (The copy has been deleted)



g. (2 pt) View all borrowing records.

The manager can view all the borrowing records for all users. The image below shows an example of the loan records. The records are displayed in the order of user ID, book ID, borrow date, due date, and return date.

localhost:3000/view_loans.html

4	2344	2023-07-23T22:55:40.435Z	2023-08-05T00:00:00.000Z	2023-07-23T22:55:49.511Z
9	9	2023-07-24T17:14:12.620Z	2023-07-31T17:14:12.620Z	2023-07-24T17:14:29.044Z
9	3000	2023-07-25T17:09:41.051Z	2023-08-01T17:09:41.051Z	2023-07-27T17:26:57.096Z
11	3000	2023-07-25T17:09:58.428Z	2023-08-01T17:09:58.428Z	Not returned
3	1	2023-07-27T00:33:43.001Z	2023-08-03T00:33:43.001Z	Not returned
8	8	2023-07-27T00:41:53.838Z	2023-08-03T00:41:53.838Z	2023-07-27T00:42:05.967Z
8	19	2023-07-27T05:56:27.955Z	2023-08-03T05:56:27.955Z	Not returned
8	24	2023-07-27T05:56:48.027Z	2023-08-03T05:56:48.027Z	2023-07-27T05:56:55.052Z
8	24	2023-07-27T05:56:48.901Z	2023-08-03T05:56:48.901Z	2023-07-28T02:09:49.397Z
9	3000	2023-07-27T16:49:54.506Z	2023-08-03T16:49:54.506Z	Not returned
9	3000	2023-07-27T17:26:46.283Z	2023-08-03T17:26:46.283Z	Not returned
9	8	2023-07-28T01:34:25.669Z	2023-08-04T01:34:25.669Z	Not returned
6	24	2023-07-28T01:41:19.673Z	2023-08-04T01:41:19.673Z	2023-07-28T02:14:54.514Z



h. (2 pt) Extend the due date for a borrowing record.

To extend the due date of a borrowing record, we will need to input the loan ID and select the new due date. As shown below, I will be extending the due date for the following record:

```
_id: ObjectId('64c2a8d6e0f29013cddcab80')
userId: 9
bookId: 3000
borrowDate: 2023-07-27T17:26:46.283+00:00
dueDate: 2023-08-03T17:26:46.283+00:00
__v: 0
```

Extend Due Date

Loan ID:

New Due Date:

Extend Due Date

When the manager clicks the Extend Due Date button, a pop-up will appear that says “Due date extended successfully”. The below image shows the extended date.

```
_id: ObjectId('64c2a8d6e0f29013cddcab80')
userId: 9
bookId: 3000
borrowDate: 2023-07-27T17:26:46.283+00:00
dueDate: 2023-08-14T00:00:00.000+00:00
__v: 0
```



3. Develop a web-based software application for book search:

The user or the manager should first login with their credentials in order to see the below things;

- (5 pt) Allow users to search based on book title, author, or publisher.

Search for a Book

Title:

Author:

Publisher:

The users can search based on book title, author or publisher. The user can enter using any similar words as it is text similarity.

The below displayed are the results for the author “tay”.

A screenshot of a web browser window displaying search results for books by Taylor Swift. The URL in the address bar is "localhost:3000/search.html".

Title: Reputation
Description: An Album
Price: 60
Author: Taylor Swift
Publisher: TS
Copies: 4
[Borrow](#)

Title: Eras Tour
Description: Tour by Taylor Swift
Price: 100
Author: Taylor Swift
Publisher: TS
Copies: 7
[Borrow](#)

Title: Evermore
Description: Like its predecessor, Evermore is also a product of remote collaboration and virtual communication.
Price: 100
Author: Taylor Swift
Publisher: Republic Records LLC



- b. (3 pt) Calculate the relevance of search results based on how many times each book is borrowed.

The image below shows how many times each book has been borrowed. The data is from the loans of the books that have been borrowed.

Popular Books			
Organized leadingedge product			
Author:	Amy Smith	Borrowed:	3 times
Publisher:	Nichols-Simon		
Reputation			
Author:	Taylor Swift	Borrowed:	2 times
Publisher:	TS		
Exclusive foreground protocol			
Author:	Louis Davis	Borrowed:	2 times
Publisher:	Yates Inc		
Perfect			
Author:	One Direction	Borrowed:	2 times
Publisher:	one D		

- c. (3 pt) Calculate the relevance of search results using text similarity provided by a vector database and word vectors.

The below is the search results using text similarity and words.

Search for a Book

Title:	<input type="text" value="centric"/>
Author:	<input type="text" value="Author"/>
Publisher:	<input type="text" value="Publisher"/>
<input type="button" value="Search"/>	

The results for the word centric are as in the image below:



Title: User-centric tangible methodology

Description: Cup bring outside off not wife usually. To such citizen me age pass would. Much member ask. Us house ago any. Truth describe factor determine nearly.

Price: 51

Author: Joshua Alvarez

Publisher: Gonzalez-Coffey

Copies: 5

[Borrow](#)

Title: Ameliorated eco-centric open system

Description: Woman require ever lay right. Live there artist wife sea green. System few dream late interview old pass. Part right maybe four. When door must central knowledge start television.

Price: 37

Author: Steven Allison

Publisher: Wilson-Patterson

Copies: 2

[Borrow](#)