

BAHRIA UNIVERSITY KARACHI CAMPUS

DEPARTMENT OF COMPUTER SCIENCE



Bahria University
Discovering Knowledge

Database Management System Lab (CSL-220)

BS CS 4B – Spring 2024

PROJECT NAME:

LIBRARY MANAGEMENT SYSTEM

Submitted by:

Mohammad Soban (02-134222-010)

Sakina Yameen (02-134222-048)

Abiha Jawaaid (02-134222-077)

Submitted To:

Miss Hareem Fatima

Sir Malik M Ali

Submitted On:

13/June/2024

Acknowledgment

I want to extend my heartfelt gratitude to my group mates for their dedication, hard work, and collaboration in developing our Library Management System project in DBMS. Your teamwork and support were instrumental in overcoming challenges and ensuring the project's success.

I also want to express my sincere thanks to our teachers for their invaluable guidance and support throughout this project. Your expertise and encouragement have been invaluable in shaping our project and helping us navigate through complexities.

Together, with the support of our teachers and the dedication of our team, we were able to accomplish our goals successfully.

Thank you all for your unwavering commitment and contributions throughout this journey.

Abstract

The Library Management System is a comprehensive and user-friendly platform meticulously designed to streamline library operations for both administrators and students. Within the system, the Admin domain offers robust features for efficient management of books and student records.

Administrators have the capability to perform various tasks such as adding new books to the library database, updating existing entries, and deleting outdated information. They can also issue alerts for fines and manage student records effectively. Additionally, the Admin domain allows administrators to approve book requests swiftly, ensuring smooth circulation processes.

On the other hand, the Student domain provides students with intuitive tools to request and manage books seamlessly. Students can easily browse through available titles, request books, and track their requests. It serves as a platform that not only simplifies administrative tasks but also empowers students to interact with library resources more effectively.

Overall, the system is designed to meet the needs of both administrators and students, fostering a more efficient and user-friendly library experience.

Introduction

The Library Management System (LMS) has been meticulously developed to revolutionize library operations and elevate user experience to new heights. This robust system consists of two primary domains: Admin and Student, each tailored to fulfill specific needs.

Within the Admin domain, administrators have comprehensive control over library resources and student records. They can efficiently manage books by adding new entries, updating information, and removing outdated records. Additionally, the Admin domain empowers administrators to promptly approve book requests, facilitating seamless circulation processes.

In the Student domain, students are provided with intuitive tools to interact with library resources. They can easily request books, explore the extensive catalog, manage borrowed books, and keep track of fines incurred. This domain is designed to empower students in their academic pursuits while promoting responsible library usage.

The primary goal of this system is to establish an organized and efficient library environment that benefits both administrators and students alike. By streamlining operations and enhancing accessibility, the Library Management System aims to optimize the library experience for all users.

Table of Content

1	Acknowledgement	2
2	Abstract	3
3	Introduction	4
4	Problem Statement	6
5	Existing Systems	6
6	Objectives and Goals	7
7	Project Scope	7
8	WorkFlow	9
9	Overview of the Project	18
10	Tools and Technologies	18
11	Project Features	18
12	DBMS Implemented Concepts	20
13	Some GUI Output	36
14	Future Work	40
15	Conclusion	41
16	References	42

Problem Statement

In many educational institutions, libraries face significant challenges in managing their resources efficiently. Traditional methods of handling book inventories, student records, and book requests often lead to errors, delays, and inefficiencies. Administrators struggle with maintaining accurate records, issuing timely alerts for fines, and managing book requests effectively. Meanwhile, students encounter difficulties in accessing updated information on book availability, tracking borrowed books, and managing fines. These issues can result in decreased library usage and a diminished overall user experience. Therefore, there is a need for a robust, automated Library Management System that streamlines these processes, enhances operational efficiency, and improves user satisfaction for both administrators and students.

Existing Systems

Our Library Management System (LMS) stands out from existing systems in several key ways:

1. **User-Centric Design:** Unlike many existing systems, our LMS prioritizes user experience. The system's design focuses on simplicity and ease of use, ensuring that users can quickly and efficiently access the information they need.
2. **Comprehensive Functionality:** While some existing systems may offer similar features, our LMS provides a comprehensive suite of functionalities tailored to the specific needs of library administrators and students. From book management and user administration to fine tracking and book requests, our system covers all aspects of library management in one cohesive platform.
3. **Real-Time Updates:** Unlike traditional and legacy systems that may rely on manual updates, our LMS utilizes real-time data synchronization to ensure that information on book availability, student records, and fines is always up-to-date. This helps reduce errors and delays in library operations.
4. **Customization Options:** Our LMS offers customization options that allow institutions to tailor the system to their specific requirements. Whether it's adjusting fine policies or customizing the user interface, administrators have the flexibility to adapt the system to their unique needs.
5. **Accessibility:** Our LMS prioritizes accessibility, ensuring that all users, regardless of their level of technological proficiency, can easily navigate the system. Features such as clear navigation menus, helpful tooltips, and accessibility options make the system inclusive for all users.

Objective

The primary objective of our Library Management System (LMS) is to modernize and streamline library operations by providing a comprehensive and user-friendly platform for administrators and students. By leveraging technology, our system aims to automate routine tasks, improve accuracy in record-keeping, enhance resource accessibility, and ultimately enhance the overall user experience within the library environment.

Goals

1. **Efficiency:** To automate manual processes and streamline administrative tasks such as book management, user administration, and fine tracking, thereby increasing operational efficiency and reducing workload for library staff.
2. **Accuracy:** To ensure the accuracy and integrity of library records by implementing real-time updates and synchronization mechanisms, minimizing errors and discrepancies in book availability, student records, and fine tracking.
3. **Accessibility:** To prioritize accessibility by designing intuitive user interfaces, providing helpful tooltips and guidance, and offering customization options to accommodate users of all technological proficiencies and accessibility needs.
4. **User Experience:** To enhance the overall user experience for both administrators and students by providing a user-centric design, intuitive navigation, and seamless integration of essential functionalities tailored to the specific needs of library patrons.
5. **Resource Management:** To optimize resource utilization by providing administrators with tools for efficient book management, including adding, updating, and deleting book records, as well as issuing alerts for fines and approving book requests in a timely manner.
6. **User Satisfaction:** To improve user satisfaction and engagement within the library community by offering a modern, efficient, and user-friendly platform that meets the diverse needs of administrators and students, fostering a more positive and productive library experience for all stakeholders.

Project Scope

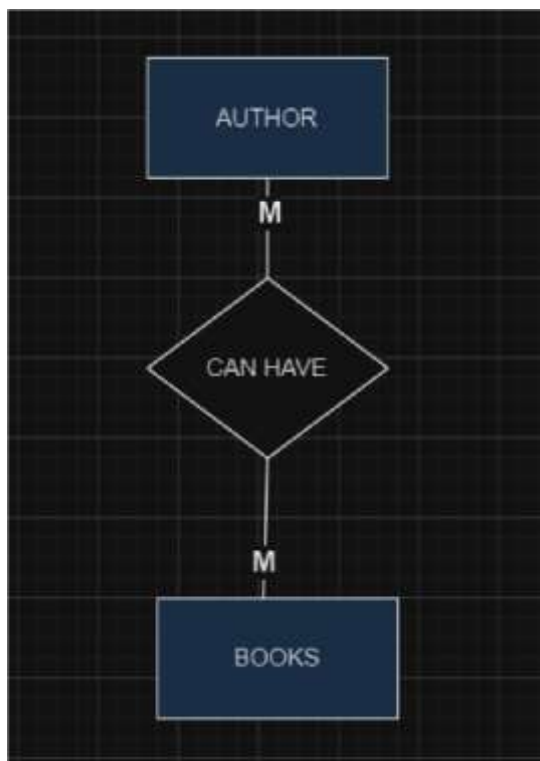
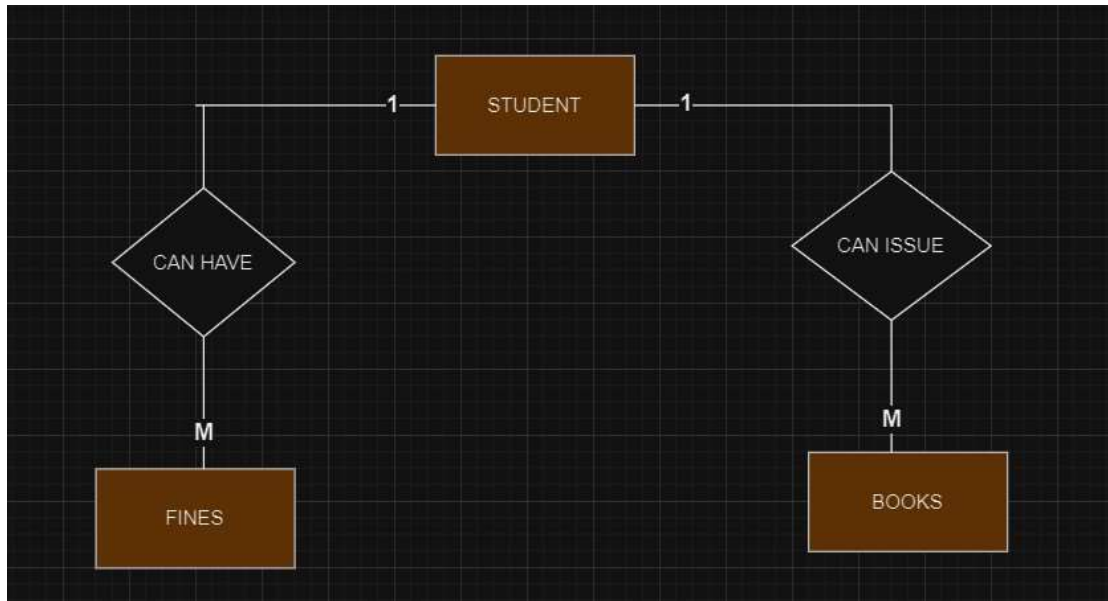
The scope of our Library Management System (LMS) project encompasses the development and implementation of a comprehensive software solution designed to facilitate efficient library operations for administrators and students. The project includes the following key components:

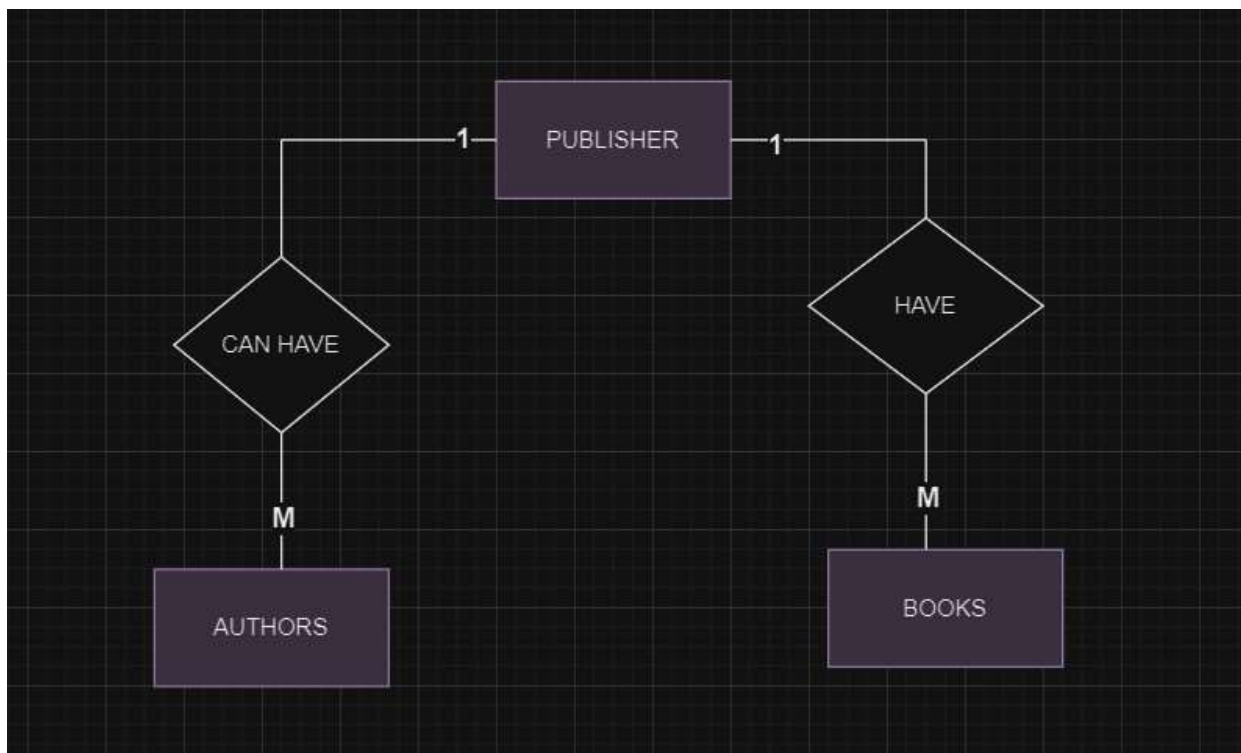
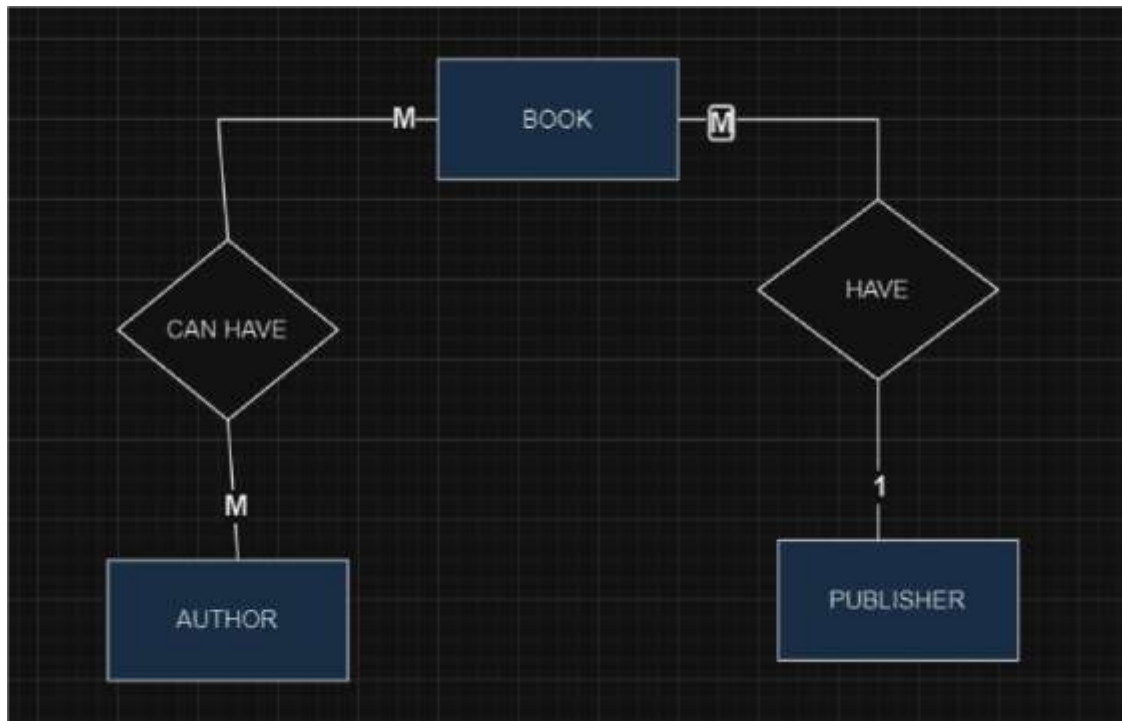
1. **Admin Domain:** The system will include functionalities for administrators to manage book inventories, student records, book requests, fines, and notifications. Admins will be able to add, update, delete, and approve book requests, as well as issue alerts for fines and manage student records.

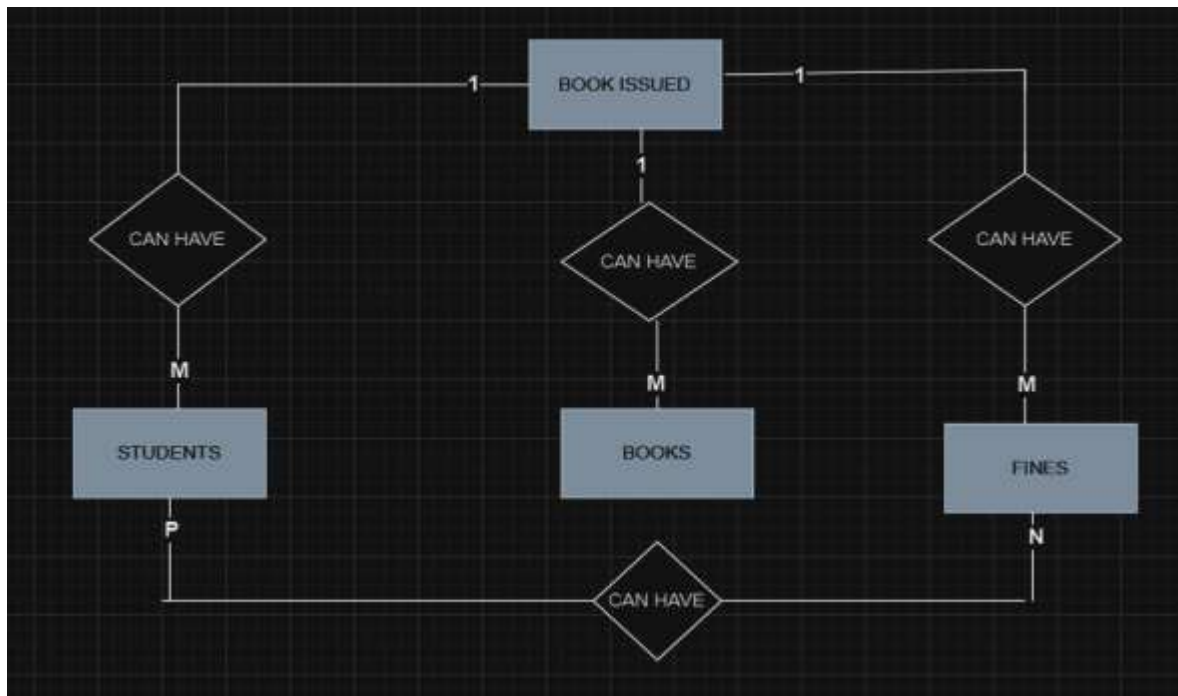
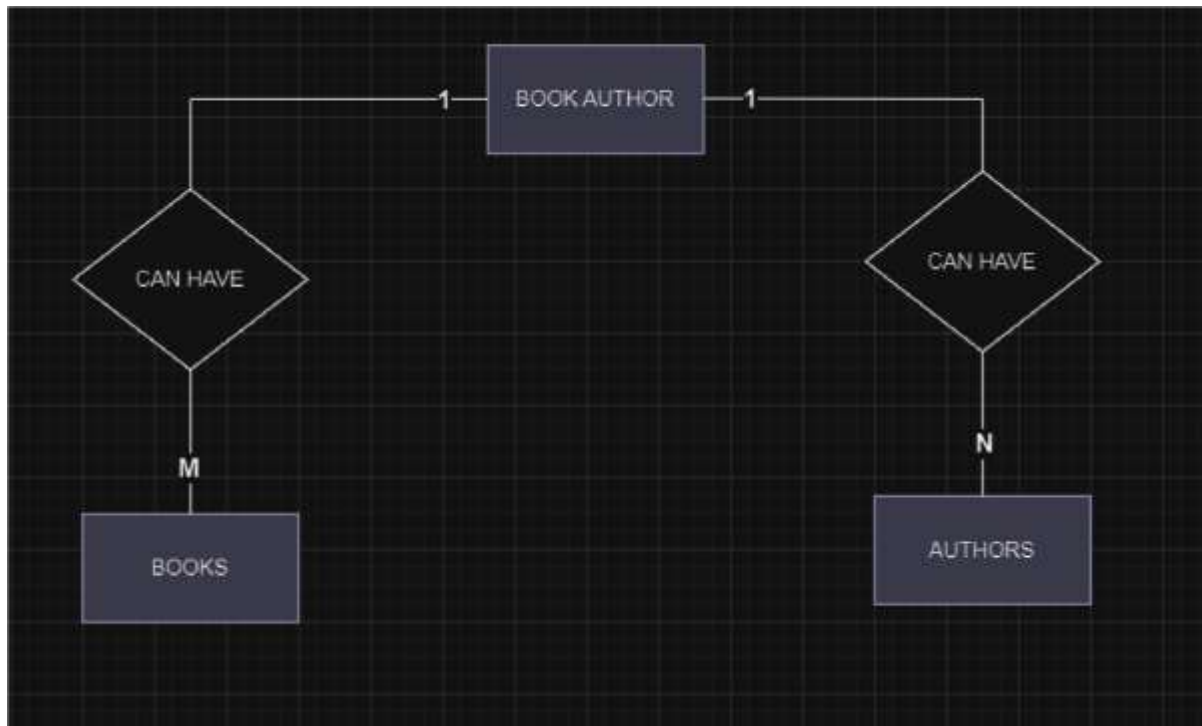
2. **Student Domain:** The system will provide students with access to features such as browsing the library catalog, requesting books, managing borrowed books through a "My Book" section, and viewing fines. Students will be able to search for books, request books, view their borrowed books, and track any fines issued by the administration.
3. **Database Management:** The system will utilize a robust database management system to store and manage library-related data, including book records, student information, fine details, and transaction logs. The database will be designed to ensure data integrity, security, and scalability, accommodating the system's growing needs over time.
4. **Integration and Customization:** The system will offer integration capabilities with external systems, such as student information systems or library catalog databases, to facilitate data exchange and interoperability. Additionally, the system will provide customization options to allow institutions to tailor the system to their specific requirements and preferences.
5. **Documentation and Training:** The project will include the development of comprehensive documentation, including user manuals, technical documentation, and training materials, to support system implementation and adoption. Training sessions will be conducted to familiarize administrators and users with the system's functionalities and usage.
6. **Deployment and Support:** The project will involve the deployment of the system in the production environment, followed by ongoing support and maintenance to address any issues, implement updates, and provide technical assistance as needed.

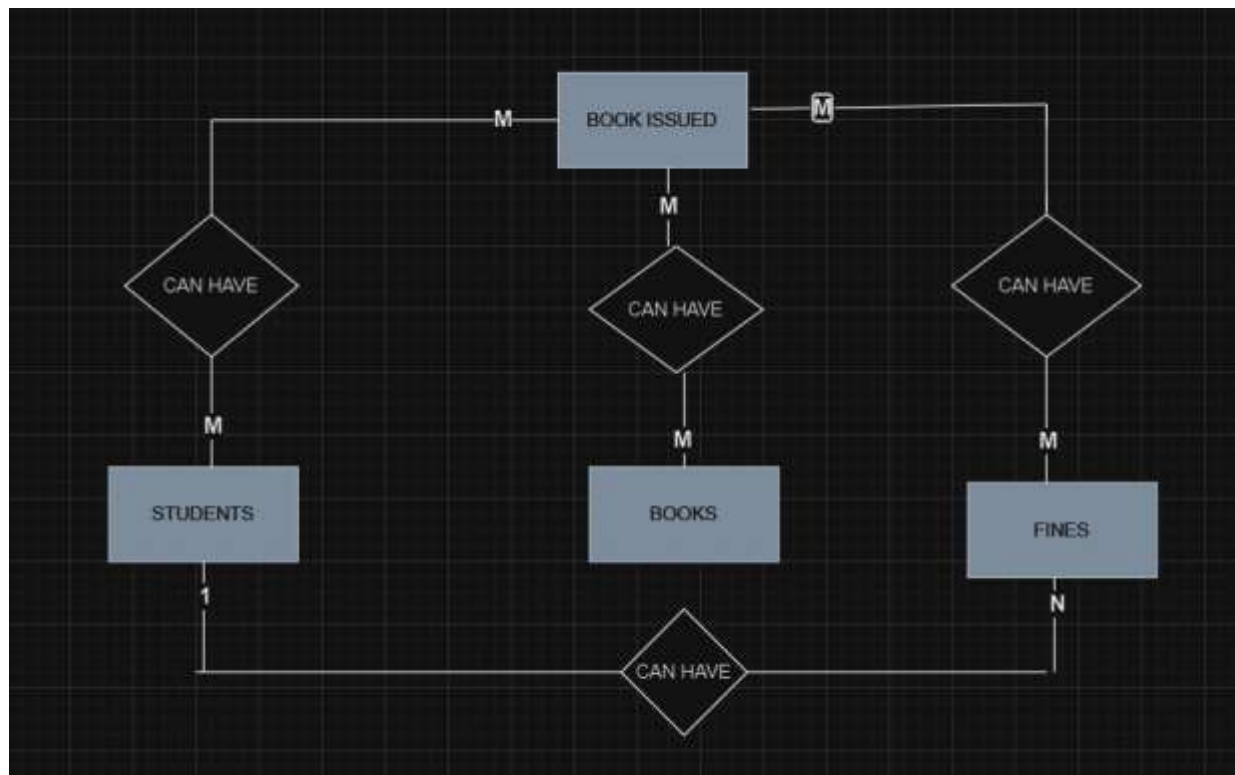
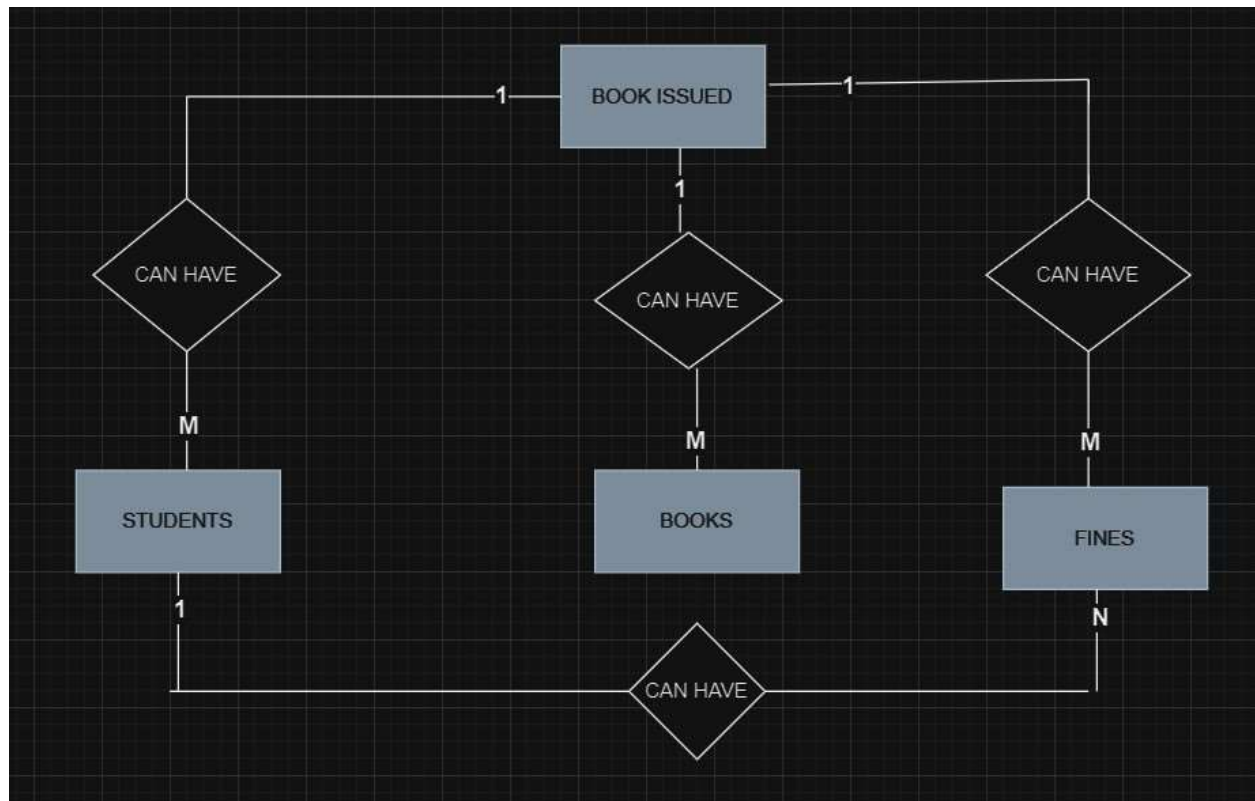
WorkFlow

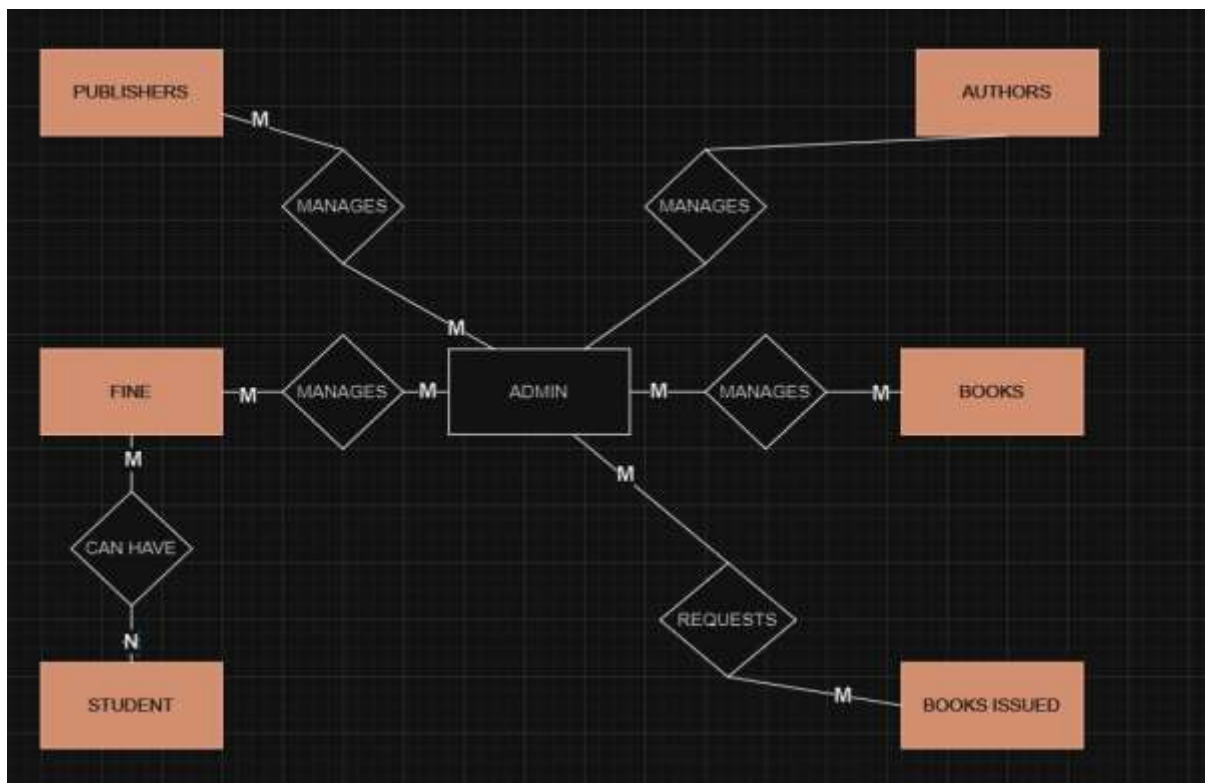
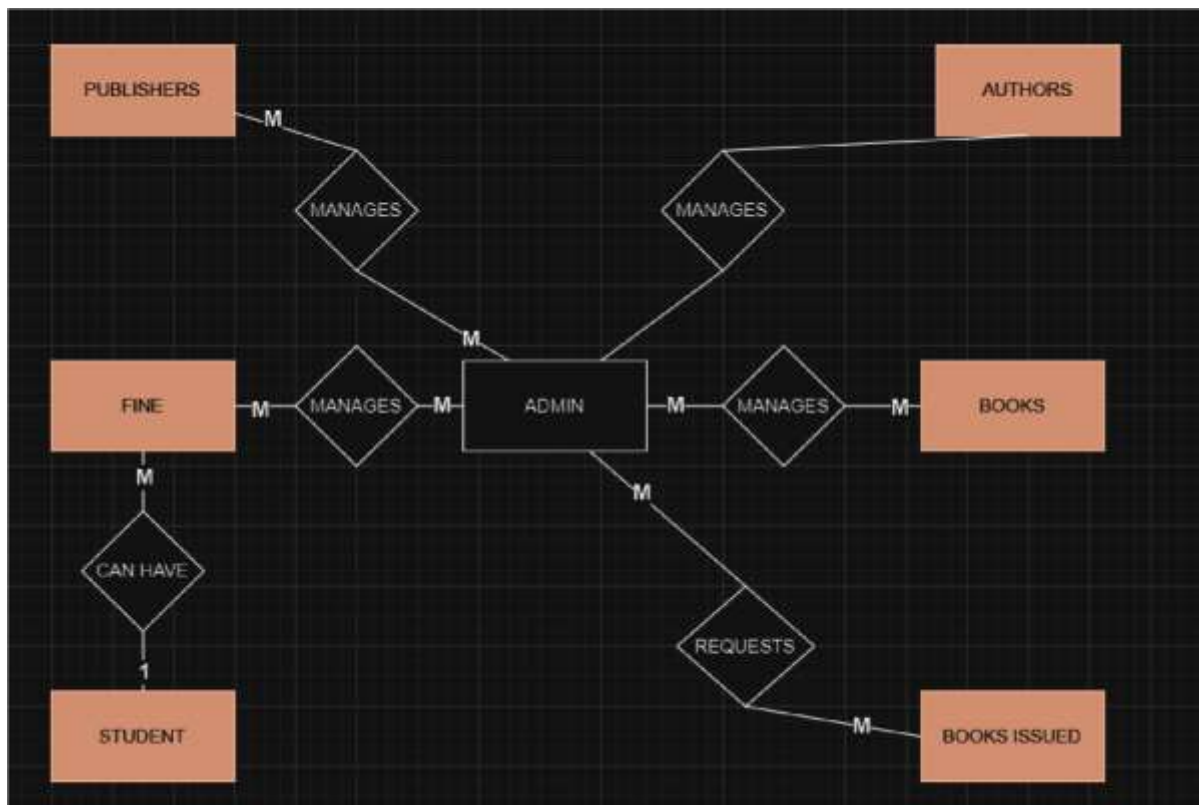
Entity Relationship Diagram (ERD's)

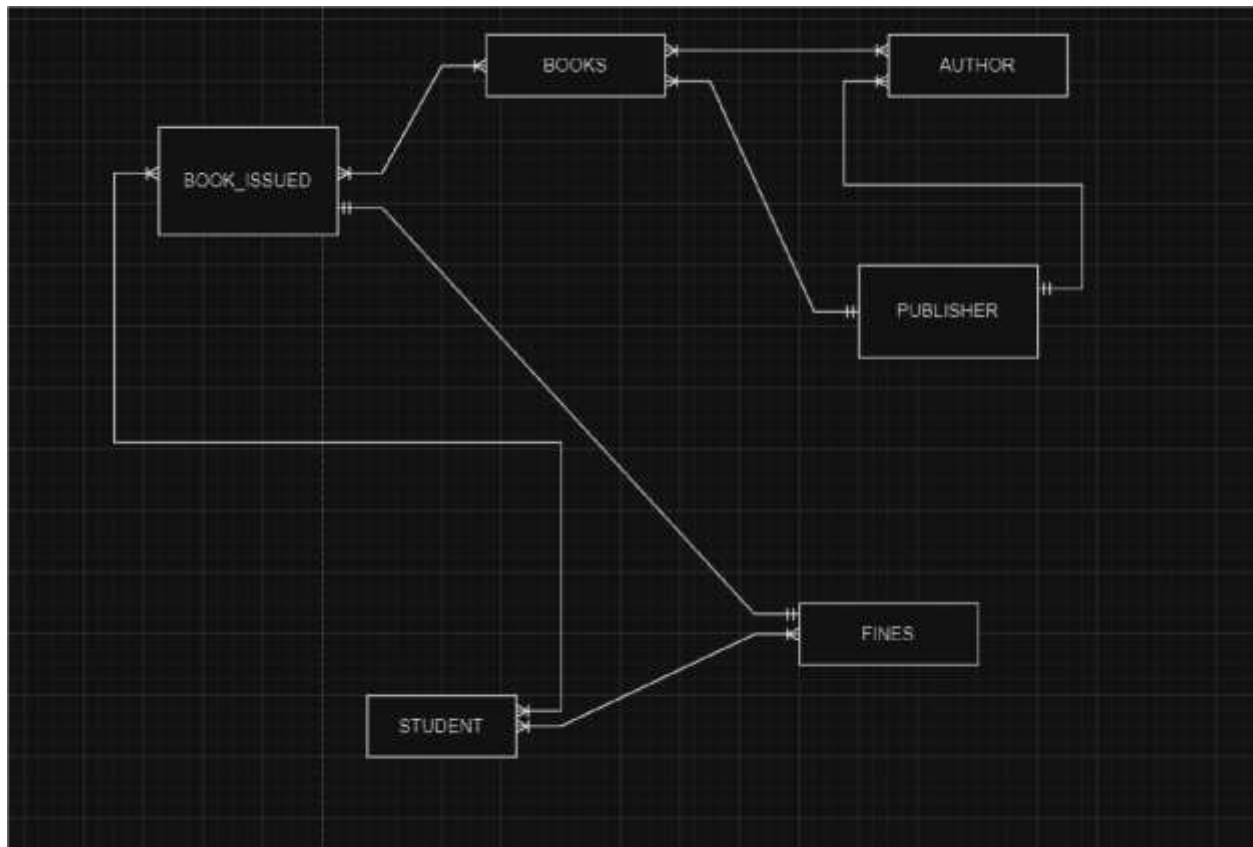






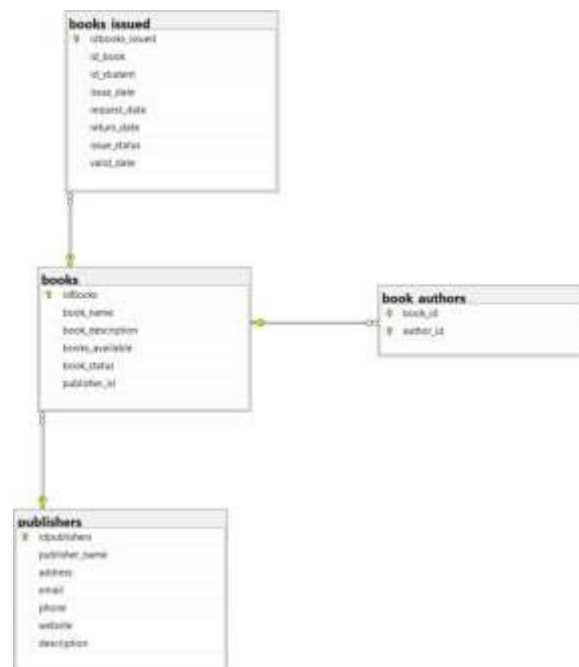




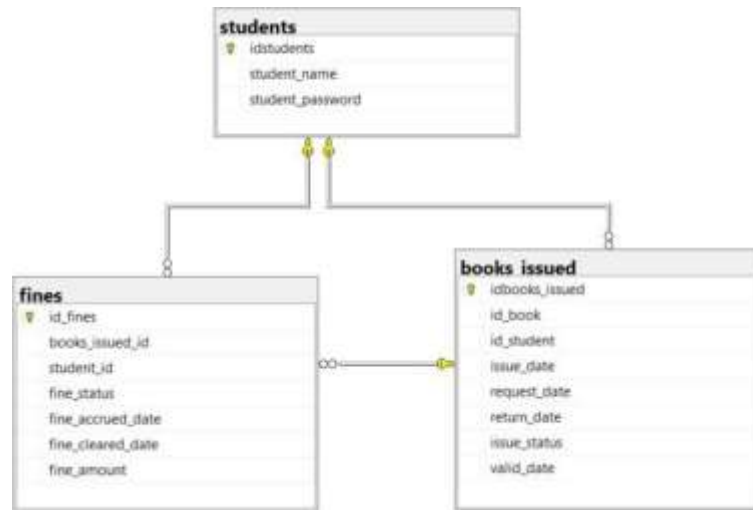


Normalized Tables up to BCNF (SQL Server Schema Diagram)

BOOKS:



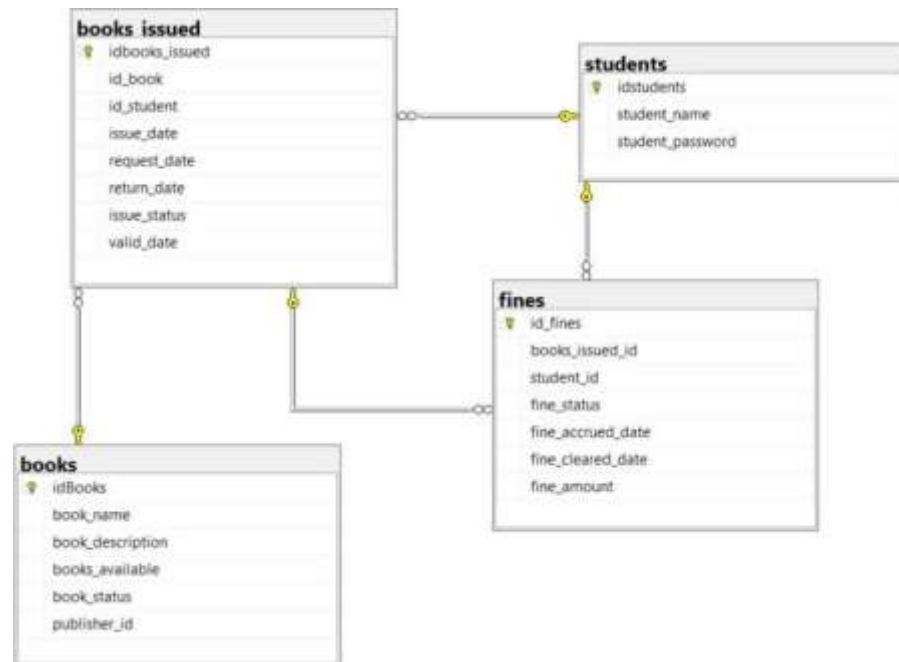
STUDENTS:



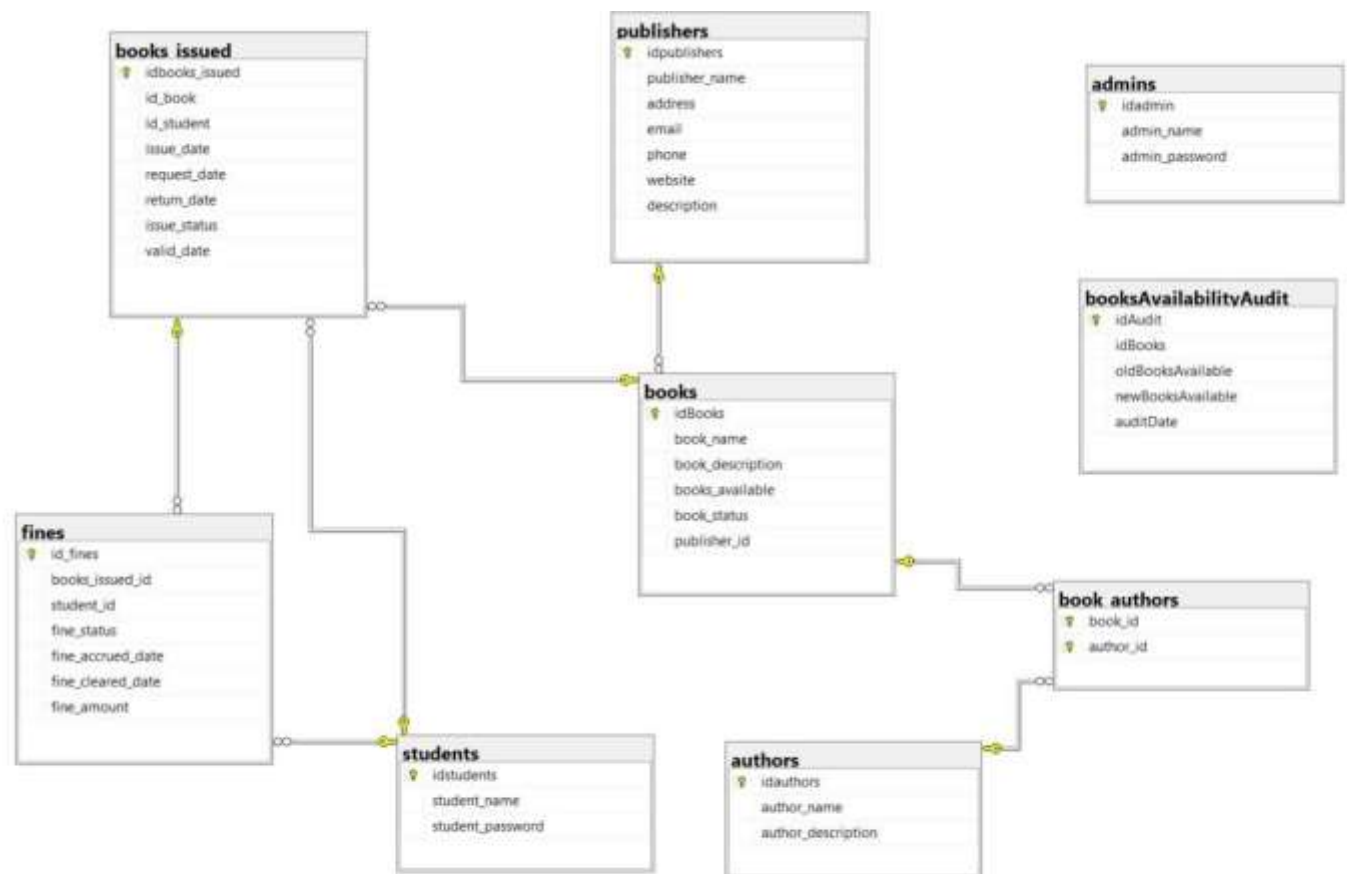
BOOK AUTHORS



FINES:



SQL Server Schema Diagram:



Overview of the Project

Library Management System encompass a range of essential features to ensure smooth and efficient operations. These include user authentication, allowing secure login with credentials, and comprehensive admin capabilities such as adding, updating, and deleting book records, managing student information, issuing fines alerts, and handling book requests. For students, functionalities cover browsing the library catalog, requesting books, managing borrowed items, and viewing fines incurred. Additionally, the system must support robust search capabilities, enabling users to find books by title, author, or category, and allow students to reserve books currently unavailable. Automated fine calculations for overdue books, email notifications for overdue items and fines, and the ability to generate detailed reports on book availability, borrowing history, and fines collected round out the system's capabilities, ensuring it meets the diverse needs of both administrators and library patrons efficiently.

Tools and Technologies

- **SQL Server Database:** Utilized for backend data storage, providing reliability and scalability.
- **Java Programming Language:** Primary language for system logic, ensuring versatility and platform independence.
- **NetBeans Integrated Development Environment (IDE):** Facilitated development tasks with its comprehensive features and user-friendly interface.
- **Java Swing:** Used to design intuitive graphical user interfaces (GUIs), enhancing user experience.

Project Features

Database Design in SQL Server:

- Create tables for authors, books, student, and so forth.
- To preserve the integrity of the data, establish relationships between the tables.
- Information from the store, including details about books, student, past borrowing activity, and more.

Authorization and Authentication of Users:

- Put in place user authentication and login procedures.
- Create several user roles with matching access levels, such as administrator and student.

Organizing Books:

- Add book record
- Delete book records.
- Update book record

Member Administration:

- Keep track of member information, such as contact and personal details.
- Control the enrollment and renewal of members.
- Maintain a record of the members' fine histories.

User interface:

- Create an interface that is easy to use for members and librarians.
- Used responsive web design to make your website accessible across a range of devices.
- Assure easy interaction and navigation.

Security Procedures:

- Put security measures in place to safeguard private information.
- Prevent common vulnerabilities by employing secure coding practices.

Main functionalities:

- ✓ Incorporating, amending, and eliminating student data.
- ✓ Handling fines and keeping track of past fines imposed.
- ✓ Preserving publisher and author information.
- ✓ Cataloging every book in the library's collection.
- ✓ Controlling access and accounts for administration.
- ✓ Keep an eye on the books that are available in the library

DBMS Implemented Concepts:

Creation of Database and Tables:

➤ Students:

```
CREATE TABLE Students (  
    student_id INT PRIMARY KEY,  
    student_name VARCHAR(50) NOT NULL,  
    student_password VARCHAR(50) NOT NULL  
);
```

	student_id	student_name	student_password
1	1	Ali	Ali123
2	2	Sara	Sara123
3	3	Ahmed	Ahmed123
4	4	Zainab	Zainab123
5	5	Hassan	Hassan123
6	6	Ayesha	Ayesha123
7	7	Umar	Umar123
8	8	Fatima	Fatima123
9	9	Bilal	Bilal123
10	10	Maryam	Maryam123
11	11	Usman	Usman123
12	12	Khadija	Khadija123
13	13	Raza	Raza123
14	14	Hina	Hina123
15	15	Salman	Salman123
16	16	Amna	Amna123
17	17	Khalid	Khalid123
18	18	Amina	Amina123
19	19	Imran	Imran123
20	20	Sadia	Sadia123

➤ Publishers:

```
CREATE TABLE Publishers (  
    publisher_id INT PRIMARY KEY,  
    publisher_name VARCHAR(50) NOT NULL,  
    address TEXT NOT NULL,  
    email VARCHAR(50) NOT NULL,  
    phone VARCHAR(15) NOT NULL,  
    website VARCHAR(50) NOT NULL,  
    description TEXT NOT NULL);
```

	publisher_id	publisher_name	address	email	phone	website	description
1	1	Oxford University Press	Karachi, Pakistan	info@oup.com	021-111-222-333	www.oup.com	Leading publisher of academic and educational ma...
2	2	Ferozsons	Lahore, Pakistan	contact@ferozsons.com	042-111-444-555	www.ferozsons.com	Publishes a wide range of books across various ge...
3	3	Sang-e-Meel Publications	Lahore, Pakistan	info@sangemeel.com	042-222-666-777	www.sangemeel.com	Specializes in publishing Urdu literature and cultura...
4	4	Penguin Random House	New York, USA	info@penguinrandomhouse.com	+1-212-555-5555	www.penguinrandomhouse.com	One of the largest English-language trade book pu...
5	5	HarperCollins	New York, USA	contact@harpercollins.com	+1-212-555-6666	www.harpercollins.com	Publishes a diverse range of fiction and non-fiction...
6	6	Simon & Schuster	New York, USA	info@simonandschuster.com	+1-212-555-7777	www.simonandschuster.com	Known for publishing bestselling authors and qualiti...
7	7	Macmillan Publishers	London, UK	info@macmillan.com	+44-20-555-8888	www.macmillan.com	Publishes educational content and academic books.
8	8	Hachette Livre	Paris, France	contact@hachette.com	+33-1-555-9999	www.hachette.com	One of the world's largest publishing groups.
9	9	Liberty Books	Karachi, Pakistan	info@libertybooks.com	021-333-444-555	www.libertybooks.com	Popular chain of bookstores and publishers in Paki...
10	10	Sang-e-Sitara Publicati...	Islamabad, Pak...	info@sangesitara.com	051-555-666-777	www.sangesitara.com	Focuses on publishing works related to Pakistani c...
11	11	Bloomsbury Publishing	London, UK	info@bloomsbury.com	+44-20-777-8888	www.bloomsbury.com	A leading independent publishing house.
12	12	Macmillan Education	London, UK	info@macmillaneducation.com	+44-20-999-8888	www.macmillaneducation.com	Specializes in educational books and materials.
13	13	Random House	New York, USA	info@randomhouse.com	+1-212-777-8888	www.randomhouse.com	Publishes fiction and non-fiction books.
14	14	Puffin Books	London, UK	info@puffinbooks.com	+44-20-666-8888	www.puffinbooks.com	Focuses on children's literature.
15	15	Scholastic Corporation	New York, USA	info@scholastic.com	+1-212-999-8888	www.scholastic.com	Publishes educational materials and children's boo...

➤ Books:

```
CREATE TABLE Books (
    book_id INT PRIMARY KEY,
    book_name VARCHAR(50) NOT NULL,
    book_description TEXT NOT NULL,
    book_available INT NOT NULL,
    book_status VARCHAR(20) NOT NULL,
    publisher_id INT,
    FOREIGN KEY (publisher_id) REFERENCES Publishers(publisher_id)
);
```

	book_id	book_name	book_description	book_available	book_status	publisher_id
1	1	The Alchemist	A novel by Paulo Coelho	10	Available	1
2	2	To Kill a Mockingbird	A novel by Harper Lee	8	Available	2
3	3	1984	A novel by George Orwell	12	Available	3
4	4	The Kite Runner	A novel by Khaled Hoss...	5	Available	4
5	5	The Da Vinci Code	A novel by Dan Brown	9	Available	5
6	6	Pride and Prejudice	A novel by Jane Austen	7	Available	6
7	7	The Trial	A novel by Franz Kafka	4	Available	7
8	8	Moby Dick	A novel by Herman Mel...	6	Available	8
9	9	Brave New World	A novel by Aldous Huxley	11	Available	1
10	10	The Great Gatsby	A novel by F. Scott Fitz...	10	Available	2
11	11	War and Peace	A novel by Leo Tolstoy	3	Available	3
12	12	Crime and Punish...	A novel by Fyodor Dost...	5	Available	4
13	13	One Hundred Yea...	A novel by Gabriel Garc...	8	Available	5
14	14	Anna Karenina	A novel by Leo Tolstoy	7	Available	6
15	15	The Catcher in the...	A novel by J.D. Salinger	6	Available	7
16	16	The Lord of the Ri...	A novel by J.R.R. Tolkien	15	Available	11
17	17	Harry Potter and t...	A novel by J.K. Rowling	14	Available	12
18	18	The Hunger Games	A novel by Suzanne Co...	13	Available	13
19	19	The Cat in the Hat	A children's book by Dr...	11	Available	14
20	20	Matilda	A children's book by Ro...	10	Available	15

➤ BooksIssued:

```
CREATE TABLE BooksIssued (
    book_issued_id INT PRIMARY KEY,
    book_id INT NOT NULL,
    student_id INT NOT NULL,
    issue_date DATE NOT NULL,
    request_date DATE NOT NULL,
    return_date DATE,
    issue_status VARCHAR(20) NOT NULL,
```

```

valid_date DATE,
FOREIGN KEY (book_id) REFERENCES Books(book_id),
FOREIGN KEY (student_id) REFERENCES Students(student_id)

```

);

	book_issued_id	book_id	student_id	issue_date	request_date	return_date	issue_status	valid_date
1	1	3	6	2024-05-01	2024-04-25	2024-05-30	Issued	2024-06-01
2	2	4	8	2024-05-02	2024-04-26	2024-05-31	Issued	2024-06-02
3	3	1	10	2024-05-03	2024-04-27	2024-06-01	Issued	2024-06-03
4	4	6	12	2024-05-04	2024-04-28	2024-06-02	Issued	2024-06-04
5	5	8	14	2024-05-05	2024-04-29	2024-06-03	Issued	2024-06-05
6	6	2	5	2024-05-06	2024-04-30	2024-06-04	Issued	2024-06-06
7	7	5	7	2024-05-07	2024-05-01	2024-06-05	Issued	2024-06-07
8	8	7	9	2024-05-08	2024-05-02	2024-06-06	Issued	2024-06-08
9	9	10	11	2024-05-09	2024-05-03	2024-06-07	Issued	2024-06-09
10	10	12	13	2024-05-10	2024-05-04	2024-06-08	Issued	2024-06-10
11	11	16	16	2024-06-11	2024-06-01	NULL	Issued	2024-06-12
12	12	17	17	2024-06-12	2024-06-02	NULL	Issued	2024-06-13
13	13	18	18	2024-06-13	2024-06-03	NULL	Issued	2024-06-14
14	14	19	19	2024-01-25	2024-01-15	2024-03-20	Returned	2024-03-20
15	15	15	15	2024-06-10	2024-05-31	NULL	Overdue	2024-06-11
16	16	1	6	2024-06-16	2024-06-06	NULL	Issued	2024-06-17
17	17	2	7	2024-06-17	2024-06-07	NULL	Issued	2024-06-18
18	18	3	8	2024-06-18	2024-06-08	2024-06-10	Returned	2024-06-19
19	19	4	9	2024-06-19	2024-06-09	NULL	Overdue	2024-06-20
20	20	5	10	2024-06-20	2024-06-10	NULL	Overdue	2024-06-21

➤ Fines:

```

CREATE TABLE Fines (
    fine_id INT PRIMARY KEY,
    books_issued_id INT NOT NULL,
    student_id INT NOT NULL,
    fine_status VARCHAR(20) NOT NULL,
    fine_accrued_date DATE NOT NULL,
    fine_cleared_date DATE,
    fine_amount DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (books_issued_id) REFERENCES BooksIssued(book_issued_id),
    FOREIGN KEY (student_id) REFERENCES Students(student_id)

```

);

	fine_id	books_issued_id	student_id	fine_status	fine_accrued_date	fine_cleared_date	fine_amount
1	1	1	1	Pending	2024-05-15	2024-05-30	100.00
2	2	2	2	Paid	2024-05-20	2024-05-25	50.00
3	3	3	3	Pending	2024-05-25	2024-06-02	75.00
4	4	4	4	Pending	2024-06-01	2024-06-03	120.00
5	5	5	5	Paid	2024-06-05	2024-06-07	80.00
6	6	6	6	Pending	2024-06-10	2024-06-15	90.00
7	7	7	7	Paid	2024-06-15	2024-06-20	60.00
8	8	8	8	Pending	2024-06-18	2024-06-22	110.00
9	9	9	9	Pending	2024-06-02	NULL	85.00
10	10	10	10	Pending	2024-06-05	NULL	95.00
11	11	11	1	Pending	2024-06-10	NULL	100.00
12	12	12	2	Paid	2024-06-15	2024-06-18	55.00
13	13	13	3	Pending	2024-06-20	NULL	80.00
14	14	14	4	Pending	2024-06-25	NULL	125.00
15	15	15	5	Paid	2024-06-28	2024-06-30	85.00
16	16	16	16	Pending	2024-06-26	NULL	95.00
17	17	17	17	Paid	2024-06-27	2024-06-29	65.00
18	18	18	18	Pending	2024-06-28	NULL	115.00
19	19	19	19	Pending	2024-06-29	NULL	105.00
20	20	20	20	Paid	2024-06-30	2024-07-02	75.00

➤ Authors:

```
CREATE TABLE Authors (
  author_id INT PRIMARY KEY,
  author_name VARCHAR(50) NOT NULL,
  author_description TEXT NOT NULL
);
```

	author_id	author_name	author_description
1	1	Paulo Coelho	Brazilian author best known for his novel The Al...
2	2	Harper Lee	American novelist widely known for To Kill a Moc...
3	3	George Orwell	English novelist, essayist, journalist, and critic.
4	4	Khaled Hosseini	Afghan-American novelist and physician.
5	5	Dan Brown	American author best known for his thriller novels.
6	6	Jane Austen	English novelist known for her novels that critiqu...
7	7	Franz Kafka	German-speaking Bohemian writer.
8	8	Herman Melville	American novelist, short story writer, and poet.
9	9	Aldous Huxley	English writer and philosopher.
10	10	F. Scott Fitzg...	American novelist and short story writer.
11	11	Leo Tolstoy	Russian writer known for his novels War and Pe...
12	12	Fyodor Dosto...	Russian novelist, short story writer, essayist, an...
13	13	Gabriel Garcia...	Colombian novelist, short-story writer, screenwrit...
14	14	J.D. Salinger	American writer known for his widely read novel, ...

➤ BookAuthors:

```
CREATE TABLE BookAuthors (
  book_id INT,
  author_id INT,
  PRIMARY KEY (book_id, author_id),
  FOREIGN KEY (book_id) REFERENCES Books(book_id),
  FOREIGN KEY (author_id) REFERENCES Authors(author_id)
);
```

	book_id	author_id
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	11
15	15	14

➤ **Admin:**

```
CREATE TABLE Admin (  
    admin_id INT PRIMARY KEY,  
    username VARCHAR(50) NOT NULL,  
    password VARCHAR(50) NOT NULL  
);
```

	admin_id	username	password
1	1	Abiha	adminpassword1
2	2	Sakeena	adminpassword2
3	3	Soban	adminpassword3

➤ **BooksAvailabilityAudit:**

```
CREATE TABLE BooksAvailabilityAudit (  
    audit_id INT PRIMARY KEY,  
    book_id INT NOT NULL,  
    old_books_available INT NOT NULL,  
    new_books_available INT NOT NULL,  
    audit_date DATE NOT NULL,  
    FOREIGN KEY (book_id) REFERENCES Books(book_id)  
);
```

	audit_id	book_id	old_books_available	new_books_available	audit_date
1	1	1	12	10	2024-01-10
2	2	2	10	8	2024-02-12
3	3	3	15	12	2024-03-15
4	4	4	6	5	2024-04-18
5	5	5	11	9	2024-05-20
6	6	6	9	7	2024-06-22
7	7	7	5	4	2024-07-25
8	8	8	8	6	2024-08-27
9	9	9	14	11	2024-09-30
10	10	10	12	10	2024-10-05
11	11	11	5	3	2024-11-10
12	12	12	7	5	2024-12-15
13	13	13	10	8	2025-01-20
14	14	14	9	7	2025-02-25
15	15	15	8	6	2025-03-30

➤ **Reservations:**

```
CREATE TABLE Reservations (  
    reservation_id INT PRIMARY KEY,  
    book_id INT NOT NULL,  
    student_id INT NOT NULL,  
    FOREIGN KEY (book_id) REFERENCES Books(book_id),  
    FOREIGN KEY (student_id) REFERENCES Students(student_id)  
);
```


	reservation_id	book_id	student_id
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10
11	11	11	1
12	12	12	2
13	13	13	3
14	14	14	4
15	15	15	5
16	16	16	1
17	17	17	2
18	18	18	3
19	19	19	4
20	20	20	5

➤ BookReviews:

```
CREATE TABLE BookReviews (
    review_id INT PRIMARY KEY,
    book_id INT NOT NULL,
    student_id INT NOT NULL,
    rating INT CHECK (rating >= 1 AND rating <= 5),
    review_text TEXT,
    review_date DATE NOT NULL,
    FOREIGN KEY (book_id) REFERENCES Books(book_id),
    FOREIGN KEY (student_id) REFERENCES Students(student_id)
);
```

	review_id	book_id	student_id	rating	review_text	review_date
1	1	1	1	4	A timeless classic.	2024-06-01
2	2	2	2	5	Captivating and thought-provoking.	2024-06-02
3	3	3	3	3	Interesting concepts but slow-paced.	2024-06-03
4	4	4	4	5	Heart-wrenching and beautifully written.	2024-06-04
5	5	5	5	4	A thrilling adventure with historical intrigue.	2024-06-05
6	6	6	8	5	Delightful and witty portrayal of society.	2024-06-06
7	7	7	7	3	A challenging read with deep philosophical themes.	2024-06-07
8	8	8	8	4	Epic and filled with maritime adventure.	2024-06-08
9	9	9	9	3	Provocative dystopian vision of the future.	2024-06-09
10	10	10	10	5	A brilliant portrayal of the Jazz Age.	2024-06-10
11	11	11	11	4	A monumental work of historical fiction.	2024-06-11
12	12	12	12	5	Psychologically intense and morally complex.	2024-06-12
13	13	13	13	4	Magical realism at its finest.	2024-06-13
14	14	14	14	5	Captivating characters in a tragic love story.	2024-06-14
15	15	15	15	3	An insightful exploration of teenage angst.	2024-06-15
16	16	16	16	4	Fantasy masterpiece with rich mythology.	2024-06-16
17	17	17	17	5	The beginning of a magical journey.	2024-06-17
18	18	18	18	3	Intense survival adventure in a dystopian world.	2024-06-18
19	19	19	19	4	Fun and imaginative children's story.	2024-06-19
20	20	20	20	5	Empowering story of a young girl's intelligence.	2024-06-20

➤ **Categories:**

```
CREATE TABLE Categories (  
    category_id INT PRIMARY KEY,  
    category_name VARCHAR(50) NOT NULL  
);
```

	category_id	category_name
1	1	Fiction
2	2	Non-Fiction
3	3	Literature
4	4	Science Fiction
5	5	Fantasy
6	6	Mystery
7	7	Thriller
8	8	Romance
9	9	History
10	10	Biography

➤ **BookCategories:**

```
CREATE TABLE BookCategories(  
    category_id INT,  
    book_id INT  
    FOREIGN KEY (category_id) REFERENCES Categories(category_id),  
    FOREIGN KEY (book_id) REFERENCES Books(book_id)  
);
```

	category_id	book_id
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	1	11
12	2	12
13	3	13
14	4	14
15	5	15
16	6	16
17	7	17
18	8	18
19	9	19
20	10	20

➤ MyFines

```
CREATE TABLE MyFines (  
  fine_id INT,  
  student_id INT,  
  FOREIGN KEY (fine_id) REFERENCES Fines(fine_id),  
  FOREIGN KEY (student_id) REFERENCES Students(student_id)  
);
```

	fine_id	student_id
1	5	5
2	13	11

➤ MyBooks

```
CREATE TABLE MyBooks (  
  book_id INT,  
  student_id INT,  
  FOREIGN KEY (book_id) REFERENCES Books(book_id),  
  FOREIGN KEY (student_id) REFERENCES Students(student_id)  
);
```

	book_id	student_id
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	1	1
8	6	11
9	9	9
10	10	10

Views:

SQLQuery1.sql - AJ...tem (AJ\Abiha (53))*

```
-- Create a view to count books by a specific author
CREATE VIEW BooksByAuthorCount AS
SELECT a.author_id, a.author_name, COUNT(ba.book_id) AS book_count
FROM Authors a
JOIN BookAuthors ba ON a.author_id = ba.author_id
JOIN Books b ON ba.book_id = b.book_id
GROUP BY a.author_id, a.author_name;

select * from BooksByAuthorCount
```

76 %

Results Messages

	author_id	author_name	book_count
1	1	Paulo Coelho	1
2	2	Harper Lee	1
3	3	George Orwell	1
4	4	Khaled Hosseini	1
5	5	Dan Brown	1
6	6	Jane Austen	1
7	7	Franz Kafka	1
8	8	Herman Melville	1
9	9	Aldous Huxley	1
10	10	F. Scott Fitzg...	1
11	11	Leo Tolstoy	2
12	12	Fyodor Dosto...	1
13	13	Gabriel Garcia...	1
14	14	J.D. Salinger	1

```
CREATE VIEW StdPendingFines AS
SELECT f.fine_id, s.student_name, f.fine_amount
FROM Fines f
JOIN Students s ON f.student_id = s.student_id
AND fine_status = 'Pending';

select * from StdPendingFines
```

76 %

Results Messages

	fine_id	student_name	fine_amount
1	1	Ali	100.00
2	3	Ahmed	75.00
3	4	Zainab	120.00
4	6	Ayesha	90.00
5	8	Fatima	110.00
6	9	Bilal	85.00
7	10	Maryam	95.00
8	11	Ali	100.00
9	13	Ahmed	80.00
10	14	Zainab	125.00
11	16	Amna	95.00
12	18	Amina	115.00
13	19	Imran	105.00

```

CREATE VIEW booksWithCategories AS
SELECT c.category_name, b.book_id
FROM Categories c
JOIN BookCategories boc on c.category_id = boc.category_id
JOIN Books b on boc.book_id = b.book_id

select * from booksWithCategories

```

Results		Messages
	category_name	book_id
1	Fiction	1
2	Non-Fiction	2
3	Literature	3
4	Science Fiction	4
5	Fantasy	5
6	Mystery	6
7	Thriller	7
8	Romance	8
9	History	9
10	Biography	10
11	Fiction	11
12	Non-Fiction	12
13	Literature	13
14	Science Fiction	14
15	Fantasy	15
16	Mystery	16
17	Thriller	17
18	Romance	18
19	History	19
20	Biography	20

SQLQuery1.sql - AJ...tem (AJ\Abiha (53))* - < >

```
=CREATE VIEW BooksWithRatingAbove3 AS
SELECT b.book_name, b.book_description, b.book_status, br.rating
FROM books b
JOIN BookReviews br ON b.book_id = br.book_id
AND br.rating > 3

select * from BooksWithRatingAbove3
```

76 %

Results Messages

	book_name	book_description	book_status	rating
1	The Alchemist	A novel by Paulo Coelho	Available	4
2	To Kill a Mockingbird	A novel by Harper Lee	Available	5
3	The Kite Runner	A novel by Khaled Hosseini	Available	5
4	The Da Vinci Code	A novel by Dan Brown	Available	4
5	Pride and Prejudice	A novel by Jane Austen	Available	5
6	Moby Dick	A novel by Herman Melville	Available	4
7	The Great Gatsby	A novel by F. Scott Fitzgerald	Available	5
8	War and Peace	A novel by Leo Tolstoy	Available	4
9	Crime and Punishment	A novel by Fyodor Dostoevsky	Available	5
10	One Hundred Years of Solitude	A novel by Gabriel Garcia Marquez	Available	4
11	Anna Karenina	A novel by Leo Tolstoy	Available	5
12	The Lord of the Rings	A novel by J.R.R. Tolkien	Available	4
13	Harry Potter and the Philosop...	A novel by J.K. Rowling	Available	5
14	The Cat in the Hat	A children's book by Dr. Seuss	Available	4
15	Matilda	A children's book by Roald Dahl	Available	5

Functions:

```
CREATE FUNCTION GetPublisherInfoAndBooks (@publisher_id INT)
RETURNS TABLE
AS
RETURN (
    SELECT
        p.publisher_id,
        p.publisher_name,
        p.address,
        p.email,
        p.phone,
        p.website,
        b.book_id,
        b.book_name,
        b.book_description,
        b.book_status
    FROM Publishers p
    JOIN Books b ON p.publisher_id = b.publisher_id
    WHERE p.publisher_id = @publisher_id
);

select * from GetPublisherInfoAndBooks(1)
```

76 %

Results Messages

	publisher_id	publisher_name	address	email	phone	website	book_id	book_name	book_description	book_status
1	1	Oxford University Press	Karachi, Pakistan	info@oup.com	021-111-222-333	www.oup.com	1	The Alchemist	A novel by Paulo Coelho	Available
2	1	Oxford University Press	Karachi, Pakistan	info@oup.com	021-111-222-333	www.oup.com	9	Brave New World	A novel by Aldous Huxley	Available

```
CREATE FUNCTION GetBookAvailabilityStatus (@book_id INT)
RETURNS VARCHAR(20)
AS
BEGIN
    DECLARE @book_status VARCHAR(20);
    SELECT @book_status = book_status
    FROM Books
    WHERE book_id = @book_id;
    RETURN @book_status;
END;

SELECT dbo.GetBookAvailabilityStatus(2) AS BookAvailabilityStatus;
```

76 %

Results Messages

	BookAvailabilityStatus
1	Available

```

CREATE FUNCTION GetTotalFines (@student_id INT)
RETURNS DECIMAL(10, 2)
AS
BEGIN
    DECLARE @total_fines DECIMAL(10, 2);
    SELECT @total_fines = SUM(fine_amount)
    FROM Fines
    WHERE student_id = @student_id;
    RETURN @total_fines;
END;

select dbo.GetTotalFines(3) as TotalFine

```

76 %

Results Messages

	TotalFine
1	155.00

```

CREATE FUNCTION AvgRatingOfABook(@book_id INT)
RETURNS INT
AS
BEGIN
    DECLARE @AvgRating INT
    SELECT @AvgRating = AVG(rating)
    FROM BookReviews
    WHERE book_id = @book_id
    RETURN @AvgRating
END

select dbo.AvgRatingOfABook(5) as AverageRatingOfABook

```

76 %

Results Messages

	AverageRatingOfABook
1	4

```

CREATE FUNCTION GetBooksIssuedToStudent (@student_id INT)
RETURNS INT
AS
BEGIN
    DECLARE @books_issued INT;
    SELECT @books_issued = COUNT(*)
    FROM BooksIssued
    WHERE student_id = @student_id AND issue_status = 'Issued';
    RETURN @books_issued;
END;

SELECT dbo.GetBooksIssuedToStudent(9) AS BooksIssued;

```

%

Results Messages

BooksIssued
1

Procedures:

```

CREATE PROCEDURE ClearFine
    @fine_id INT,
    @fine_cleared_date DATE
AS
BEGIN
    UPDATE Fines
    SET fine_status = 'Cleared',
        fine_cleared_date = @fine_cleared_date
    WHERE fine_id = @fine_id;
END;

exec ClearFine 20, '2024-12-06'

select * from StudentFines where fine_id = 20

```

0 %

Results Messages

	fine_id	student_id	student_name	fine_status	fine_accrued_date	fine_cleared_date	fine_amount
1	20	20	Sadia	Cleared	2024-06-30	2024-12-06	75.00

SQLQuery1.sql - AJ...tem (AJ\Abiha (53))*

```

CREATE PROCEDURE BookReturned
    @book_issued_id INT,
    @return_date DATE
AS
BEGIN
    UPDATE BooksIssued
    set issue_status = 'Returned', return_date = @return_date
    where book_issued_id = @book_issued_id

    END

    exec BookReturned 12, '2024-07-02'

    select * from BooksIssued where book_issued_id = 12

```

70 %

Results Messages

	book_issued_id	book_id	student_id	issue_date	request_date	return_date	issue_status	valid_date
1	12	17	17	2024-06-12	2024-06-02	2024-07-02	Returned	2024-06-13

```

CREATE PROCEDURE UpdateBook
    @book_id INT,
    @book_name VARCHAR(50),
    @book_description TEXT,
    @book_available INT,
    @book_status VARCHAR(20),
    @publisher_id INT
AS
BEGIN
    UPDATE Books
    SET book_name = @book_name,
        book_description = @book_description,
        book_available = @book_available,
        book_status = @book_status,
        publisher_id = @publisher_id
    WHERE book_id = @book_id;

    END;

    exec UpdateBook 4, 'Thousand Splendid Suns', 'Best Seller Book By Khaled Hosseini', 0, 'Not Available', 10

    select * from Books where book_id = 4

```

70 %

Results Messages

	book_id	book_name	book_description	book_available	book_status	publisher_id
1	4	Thousand Splendid Suns	Best Seller Book By Khaled Hosseini	0	Not Available	10

SQLQuery1.sql - AJ...tem (AJ\Abiha (53))*

```
CREATE PROCEDURE AddStd
    @student_id INT,
    @student_name VARCHAR(50),
    @student_password VARCHAR(50)
AS
BEGIN
    INSERT INTO Students (student_id, student_name, student_password)
    VALUES (@student_id, @student_name, @student_password);
END;

exec AddStd 21, 'Arham', 'Arham123'

select * from Students
```

Results Messages

	student_id	student_name	student_password
1	1	Ali	Ali123
2	2	Sara	Sara123
3	3	Ahmed	Ahmed123
4	4	Zainab	Zainab123
5	5	Hassan	Hassan123
6	6	Ayesha	Ayesha123
7	7	Umar	Umar123
8	8	Fatima	Fatima123
9	9	Bilal	Bilal123
10	10	Maryam	Maryam123
11	11	Usman	Usman123
12	12	Khadija	Khadija123
13	13	Raza	Raza123
14	14	Hina	Hina123
15	15	Salman	Salman123
16	16	Amna	Amna123
17	17	Khalid	Khalid123
18	18	Amina	Amina123
19	19	Imran	Imran123
20	20	Sadia	Sadia123
21	21	Arham	Arham123

```

CREATE PROCEDURE AddFine
    @fine_id INT,
    @books_issued_id INT,
    @student_id INT,
    @fine_status VARCHAR(20),
    @fine_accrued_date DATE,
    @fine_amount DECIMAL(10, 2)
AS
BEGIN
    INSERT INTO Fines (fine_id, books_issued_id, student_id, fine_status, fine_accrued_date, fine_amount)
    VALUES (@fine_id, @books_issued_id, @student_id, @fine_status, @fine_accrued_date, @fine_amount);
END;

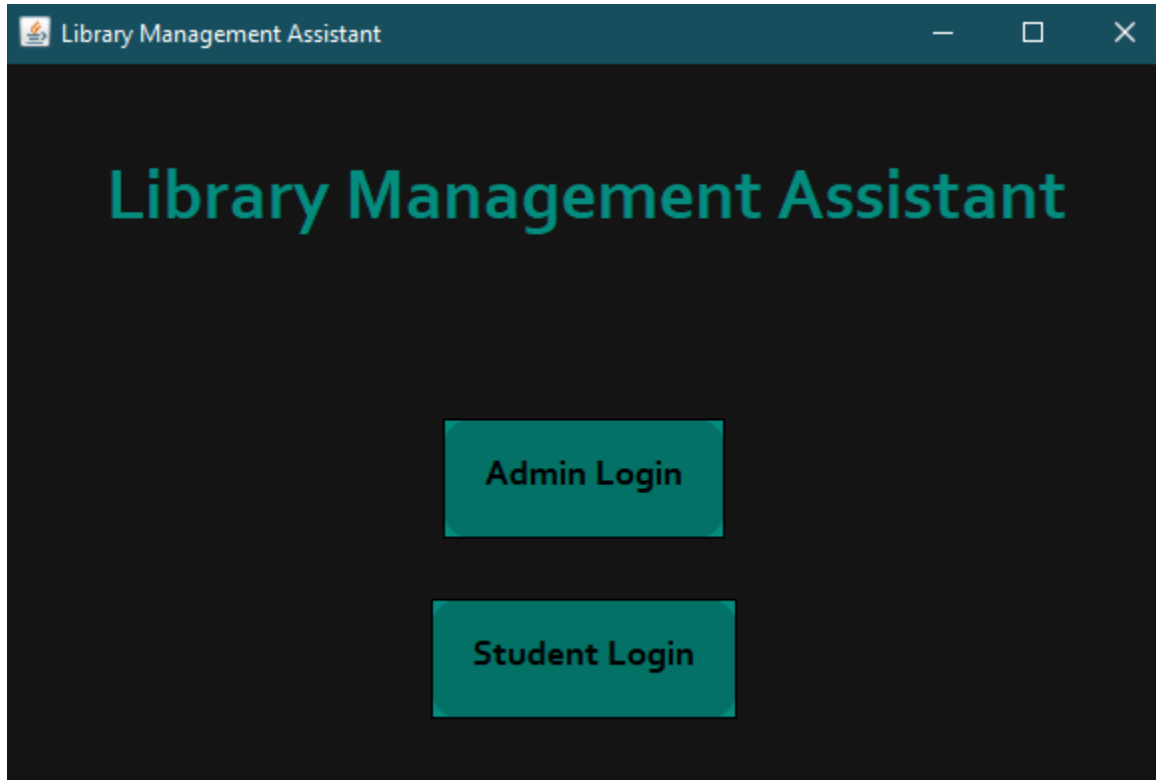
exec AddFine 21,7,5,'Pending','2024-02-1',200

```

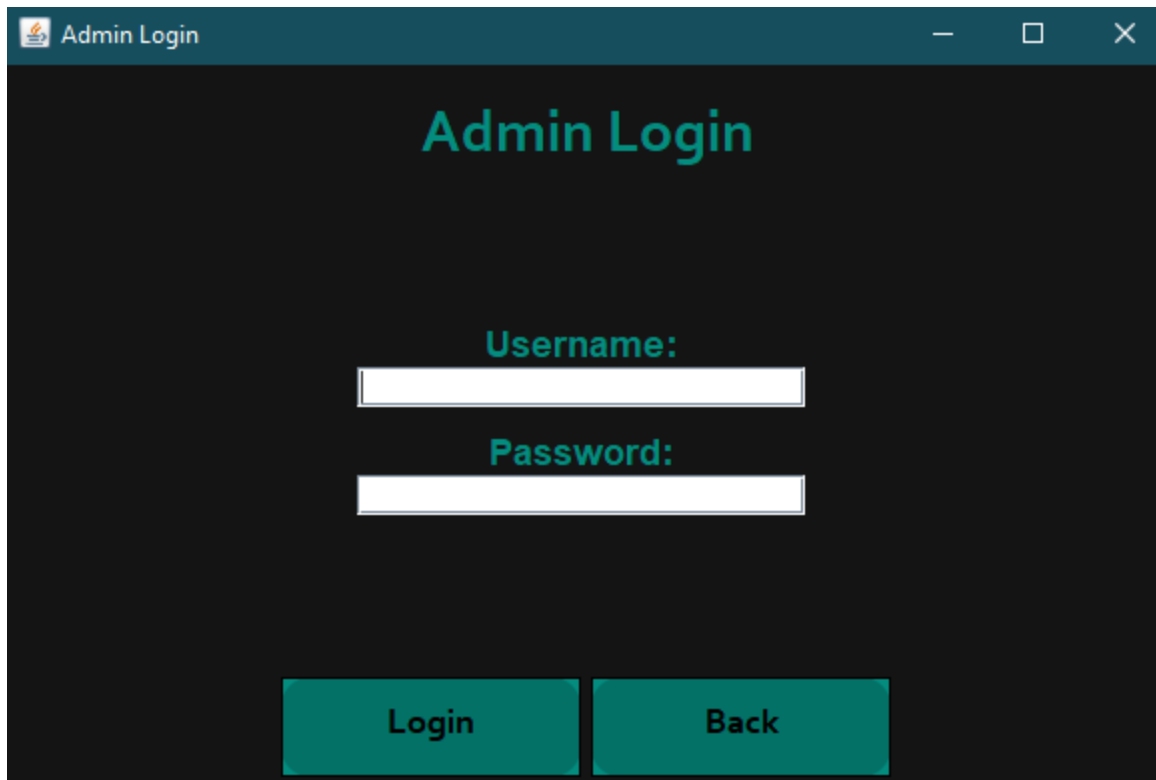
Results		Messages					
	fine_id	books_issued_id	student_id	fine_status	fine_accrued_date	fine_cleared_date	fine_amount
1	1	1	1	Pending	2024-05-15	2024-05-30	100.00
2	2	2	2	Paid	2024-05-20	2024-05-25	50.00
3	3	3	3	Pending	2024-05-25	2024-06-02	75.00
4	4	4	4	Pending	2024-06-01	2024-06-03	120.00
5	5	5	5	Paid	2024-06-05	2024-06-07	80.00
6	6	6	6	Pending	2024-06-10	2024-06-15	90.00
7	7	7	7	Paid	2024-06-15	2024-06-20	60.00
8	8	8	8	Pending	2024-06-18	2024-06-22	110.00
9	9	9	9	Pending	2024-06-02	NULL	85.00
10	10	10	10	Pending	2024-06-05	NULL	95.00
11	11	11	1	Pending	2024-06-10	NULL	100.00
12	12	12	2	Paid	2024-06-15	2024-06-18	55.00
13	13	13	3	Pending	2024-06-20	NULL	80.00
14	14	14	4	Pending	2024-06-25	NULL	125.00
15	15	15	5	Paid	2024-06-28	2024-06-30	85.00
16	16	16	16	Pending	2024-06-26	NULL	95.00
17	17	17	17	Paid	2024-06-27	2024-06-29	65.00
18	18	18	18	Pending	2024-06-28	NULL	115.00
19	19	19	19	Pending	2024-06-29	NULL	105.00
20	20	20	20	Cleared	2024-06-30	2024-12-08	75.00
21	21	7	5	Pending	2024-02-01	NULL	200.00

Some GUI Outputs:

Title Page:



Admin/Student Login:



The image shows a web browser window titled "Admin Login". The window has a dark blue header bar with the title and standard window controls (minimize, maximize, close). The main content area is black. In the center, the text "Admin Login" is displayed in a large, bold, light blue font. Below this, there are two white input fields. The first field is preceded by the label "Username:" in a light blue font. The second field is preceded by the label "Password:" in a light blue font. At the bottom of the window, there are two light blue buttons with black text: "Login" on the left and "Back" on the right.

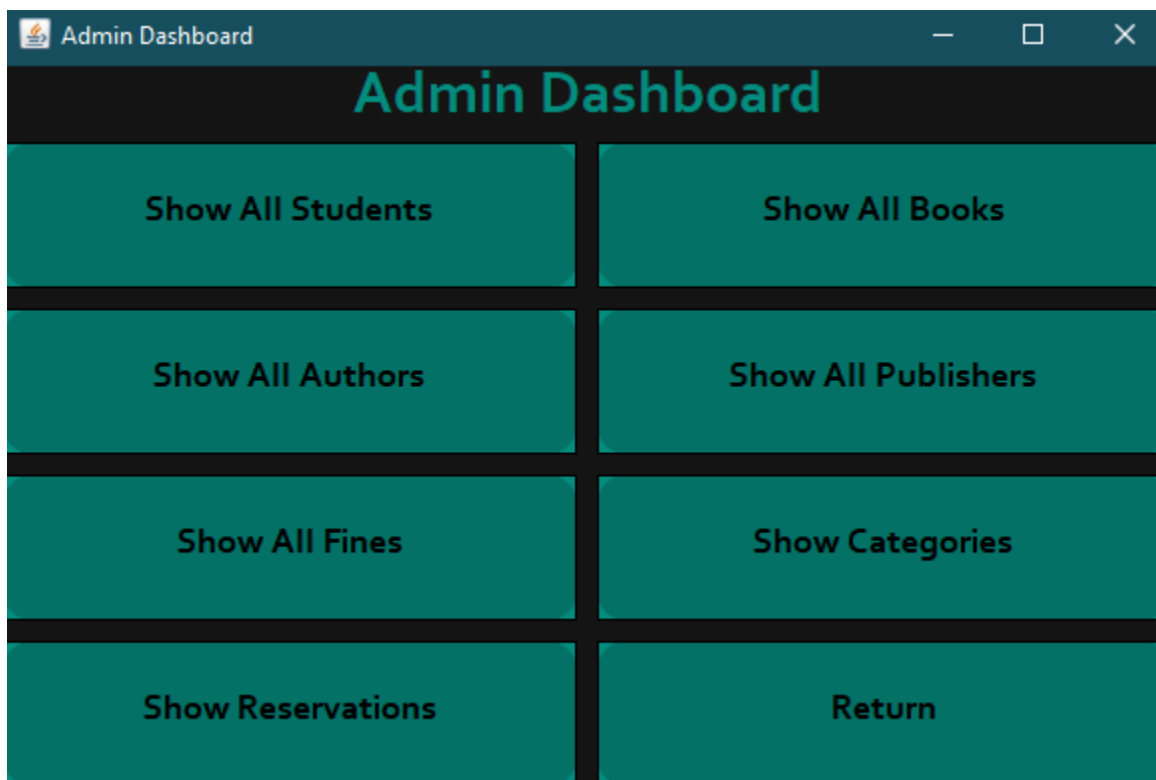
Admin Login

Username:

Password:

Login Back

Admin Dashboard:



The image shows a web browser window titled "Admin Dashboard". The window has a dark blue header bar with the title and standard window controls (minimize, maximize, close). The main content area is black. At the top, the text "Admin Dashboard" is displayed in a large, bold, light blue font. Below this, there is a grid of eight light blue buttons with black text, arranged in four rows and two columns. The buttons are: "Show All Students", "Show All Books", "Show All Authors", "Show All Publishers", "Show All Fines", "Show Categories", "Show Reservations", and "Return".

Admin Dashboard

Show All Students Show All Books

Show All Authors Show All Publishers

Show All Fines Show Categories

Show Reservations Return

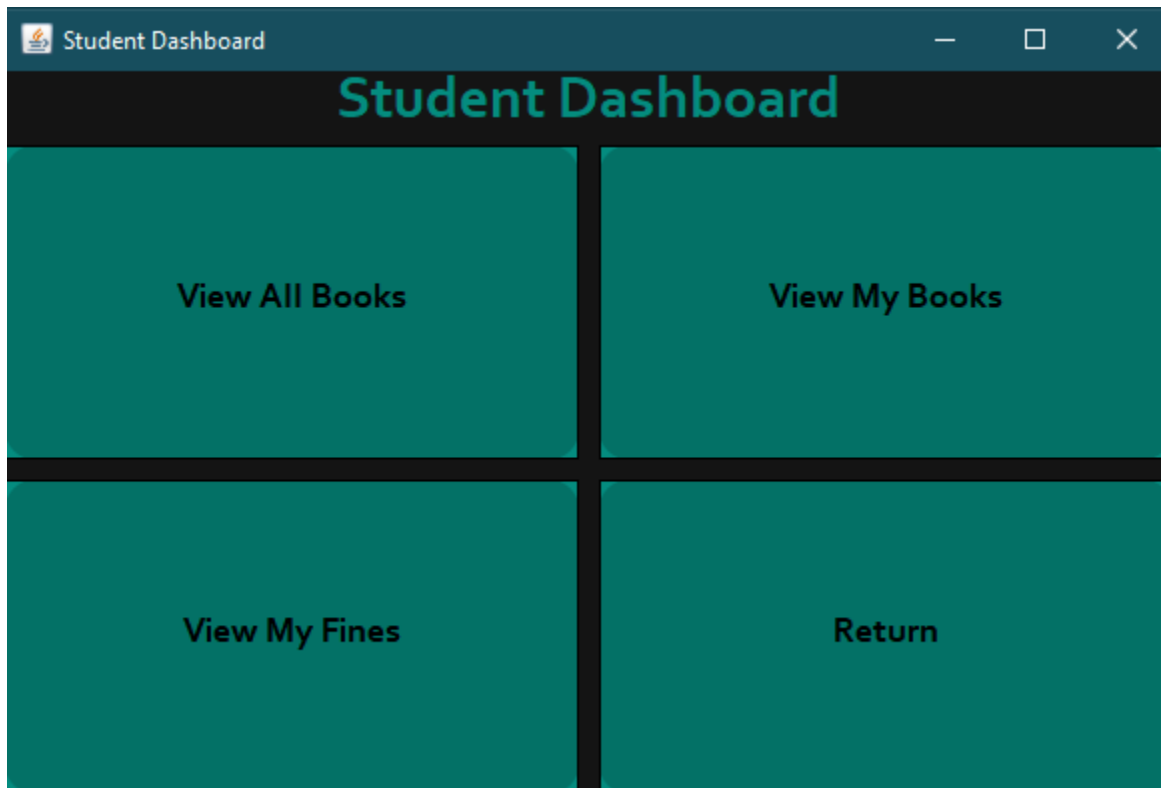
Showing All Books from Database:

Books			
Book ID	Book Name	Book Status	Publisher ID
1	The Great Gatsby	Available	1
2	To Kill a Mockingbird	Checked Out	2
3	1999	Available	7
4	Pride and Prejudice	Available	4
5	The Catcher in the Rye	Checked Out	5
6	The Hobbit	Available	6
9	Brave New World	Available	9
10	Moby-Dick	Available	10

Showing All Fines from Database:

fine_id	student_id	fine_status	fine_amount
1	1	Unpaid	5.0
2	2	Paid	3.5
3	3	Unpaid	4.75
4	4	Unpaid	3.0
5	5	Unpaid	6.0
9	9	Unpaid	3.0
10	10	Paid	2.0
12	5	Unpaid	5.0
13	11	Pending	10.0

Student Dashboard:



Showing Books Left:

The screenshot shows a web application window titled "Books". It displays a table with four columns: "Book ID", "Title", "Status", and "Publisher ID". The table contains 10 rows of data. Below the table, there is a large empty light gray rectangular area. At the bottom of the window, there are three buttons: "Back", "Request Book", and "Show Reviews".

Book ID	Title	Status	Publisher ID
1	The Great Gatsby	Available	1
2	To Kill a Mockingbird	Checked Out	2
3	1999	Available	7
4	Pride and Prejudice	Available	4
5	The Catcher in the Rye	Checked Out	5
6	The Hobbit	Available	6
9	Brave New World	Available	9
10	Moby-Dick	Available	10

Future Work

The Library Management System (LMS) project lays a solid foundation for efficient library management; however, several enhancements and expansions can be considered for future work. One potential area of development is the integration of a mobile application to provide users with on-the-go access to library resources. Additionally, incorporating advanced data analytics could help librarians gain insights into user behavior and book usage patterns, aiding in better decision-making. Another important future enhancement is the implementation of AI-driven features, such as personalized book recommendations and automated chatbots for user assistance. Expanding interoperability with other library systems and educational platforms will further enhance the system's utility and reach. Lastly, ongoing security improvements and the incorporation of more robust data privacy measures will ensure the system remains secure and compliant with evolving standards.

Conclusion

The Library Management System (LMS) project successfully addresses the core challenges of library management by providing a comprehensive and user-friendly platform for administrators and students. By streamlining administrative tasks, enhancing resource accessibility, and improving overall user experience, the LMS significantly contributes to the efficient functioning of library operations. The system's robust features, including secure user authentication, comprehensive book and student management, intuitive search functionalities, and automated fine calculations, ensure it meets the diverse needs of its users. The implementation of this system marks a significant step towards modernizing library management, fostering a more productive and engaging environment for all stakeholders.

References

1. M. Fowler, "Patterns of Enterprise Application Architecture," Addison-Wesley, 2002.
2. H. M. Deitel and P. J. Deitel, "Java: How to Program," 10th ed., Pearson Education, 2015.
3. R. C. Martin, "Clean Code: A Handbook of Agile Software Craftsmanship," Prentice Hall, 2008.
4. Oracle, "JavaFX: Getting Started with JavaFX," [Online]. Available: https://docs.oracle.com/javase/8/javafx/get-started-tutorial/get_start_apps.htm. [Accessed: 12-Jun-2024].
5. Microsoft, "SQL Server 2019: Technical Documentation," [Online]. Available: <https://docs.microsoft.com/en-us/sql/sql-server/sql-server-technical-documentation>. [Accessed: 12-Jun-2024].
6. Apache Maven, "Maven – Welcome to Apache Maven," [Online]. Available: <https://maven.apache.org/>. [Accessed: 12-Jun-2024].
7. NetBeans, "NetBeans IDE - Official IDE for Java 8," [Online]. Available: <https://netbeans.apache.org/>. [Accessed: 12-Jun-2024].
8. J. Han, M. Kamber, and J. Pei, "Data Mining: Concepts and Techniques," 3rd ed., Morgan Kaufmann, 2011.