



SCHOOL OF DIGITAL TRANSFORMATION
BACHELOR OF INFORMATION TECHNOLOGY (HONOURS)

STUDENT NAME: SRI NELSON A/L PALNERSELVAM

ID: 14043

COURSE TITLE: INTEGRATIVE PROGRAMMING

COURSE CODE: BIT6424

LECTURER NAME: MR. FEROZ

Contents

Introduction.....	3
Explanation of Code	3
Login Page:	3
User Panel:	3
Admin panel	3
Export Data to a Text File:.....	4
Object-Oriented Concepts:	4
Screenshot	5
Database	7
Test Cases.....	8
Test Cases for Login:	8
Test Cases for Delete User:	8
Test Cases for Delete Book:	8
Test Cases for Forgot Password:	8
Test Table:.....	9
Conclusion.....	9

Introduction

Library Management System is an application that is designed to manage the books and other resources of a library. This system is developed using Java Swing with NetBeans as an IDE and XAMPP as a database. The application includes two panels, a user panel, and an admin panel. The user panel provides the users with the facility to search for books and make a request for a book they want to borrow. The users can search for books based on various parameters like author, title, or category. Once they find a book they want to borrow, they can make a request for it, and the system will notify the librarian to issue the book. The admin panel provides the library staff with the facility to manage the library resources. The admin panel includes features like adding or removing a user, adding or removing a book, and managing book issues. The admin can also view the user requests and approve or reject them. The system also includes a forgot password panel and a signup panel. The forgot password panel allows the users to recover their password in case they forget it. The signup panel allows the users to register themselves with the library system and access the library resources.

Explanation of Code

Login Page:

The software begins with a login window in which users enter their username and password. For demonstration purposes, we have configured the login and password to 'admin' in this project. Swing components are used on the login page to build an easy-to-use graphical user interface (GUI) for user interaction. Object-oriented techniques such as polymorphism, inheritance, and exception handling are used to implement the login page.

User Panel:

After successful login, the user goes to the user panel where they can search for books, reserve books and save the text to a file. The user panel uses JDBC to manage data such as general ledgers and reservations. The UI also uses object-oriented concepts such as inheritance, polymorphism, and exception handling to develop a modular and extensible system. Swing components are used to create an easy-to-use graphical user interface (GUI) for user panel interaction. The user can reserve a book by entering the name of the book and the number of days for which he wants to reserve it system then inserts the reservation details into the database using JDBC

Admin panel

In a library management system, an administration panel allows librarians to manage library resources such as books, users, and loan history. The admin panel provides administrators with a graphical user interface (GUI) to perform tasks such as adding or removing books, users, and managing loan history. The admin panel is implemented using object-oriented programming concepts such as inheritance and polymorphism.

The AdminPanel class is a subclass of the Admin class, allowing it to inherit properties and methods from its parent class. The Admin class defines public properties and methods shared by all instances of the class, such as add user() and deleteUser(). The AdminPanel class adds its own unique functionality, such as addBook() and deleteBook(), that allow administrators to manage books in the library. The admin panel also has a search feature that allows administrators to search for books based on specific criteria such as title, author, or year of publication. The search function retrieves book information from a

MySQL database using SQL queries and JDBC. The retrieved data is displayed in a table so that administrators can easily browse the search results.

Another great feature of the admin panel is the ability to manage users. Administrators can add or remove users from the library management system using the `addUser()` and `deleteUser()` methods. Administrators can also view a list of all registered users and their information such as name, email address and contact phone number. For security reasons, the admin panel includes a login form where admins must enter their username and password. The application authenticates administrator credentials by checking them against a MySQL database using JDBC. Depending on the user type, it calls the `validateAdmin` method of the `UserValidator` class to verify the administrator's credentials against the database. If the credentials are valid, the administration panel opens and administrators can manage library resources.

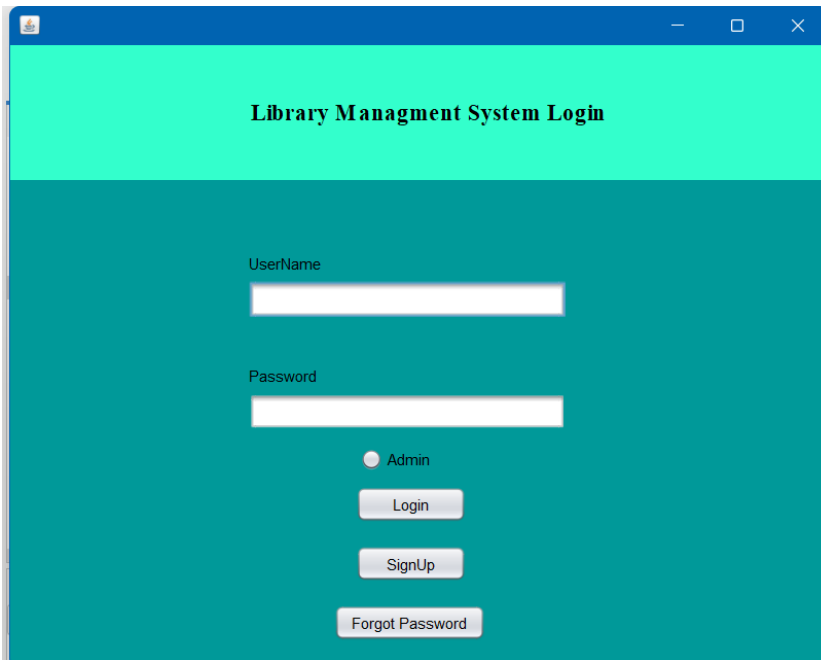
Export Data to a Text File:

Users and administrators must be able to export available data to text files. We have implemented this functionality in the user panel and admin panel. When the user clicks the export button, the system uses the `PrintWriter` class to save the book information to a text file. The system also handles any exceptions that may occur when saving a text file using exception handling and stores the book's recent borrowing history in the admin panel.

Object-Oriented Concepts:

In this project we use object -oriented concepts, such as inheritance, polymorphism and abstraction to create modular and expanded systems. Inheritance of subclasses used to create the main class (library), which allows subclasses to inherit the attributes and methods of the parenting class. Polymorphism is used in the `BookDetailsRetriever` class so that the `retrieveBookDetails` method can accept different types of input parameters and perform different processing depending on the type. Abstractions are used to define common behavior that can be shared by multiple classes. For example, the authentication interface can define login and logout methods implemented by the `Admin` and `User` classes. This allows for greater code reuse and reduces duplication in the system.

Screenshot



A screenshot of a web browser window titled "Library Management System Login". The window has a blue header bar. Below the header, the title "Library Management System Login" is centered in bold black text. The main content area has a teal background. It contains two white input fields for "UserName" and "Password". Below the password field is a radio button labeled "Admin". At the bottom, there are three buttons: "Login", "SignUp", and "Forgot Password".

Library Management System Login

UserName

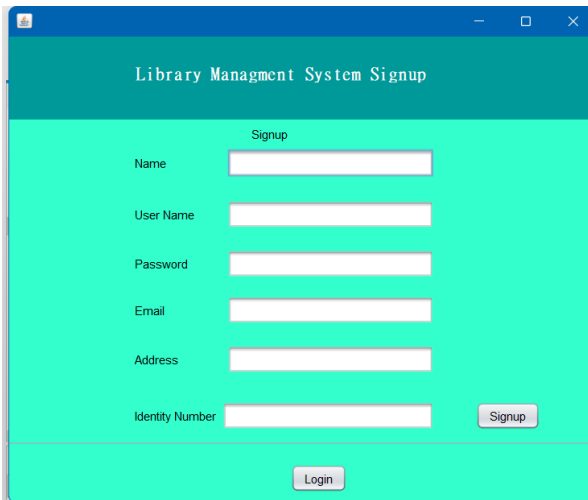
Password

☐ Admin

Login

SignUp

Forgot Password



A screenshot of a web browser window titled "Library Management System Signup". The window has a blue header bar. Below the header, the title "Library Management System Signup" is centered in bold black text. The main content area has a teal background. It contains a "Signup" section with six white input fields for "Name", "User Name", "Password", "Email", "Address", and "Identity Number". A "Signup" button is located to the right of the "Identity Number" field. At the bottom of the window, there is a "Login" button.

Library Management System Signup

Signup

Name

User Name

Password

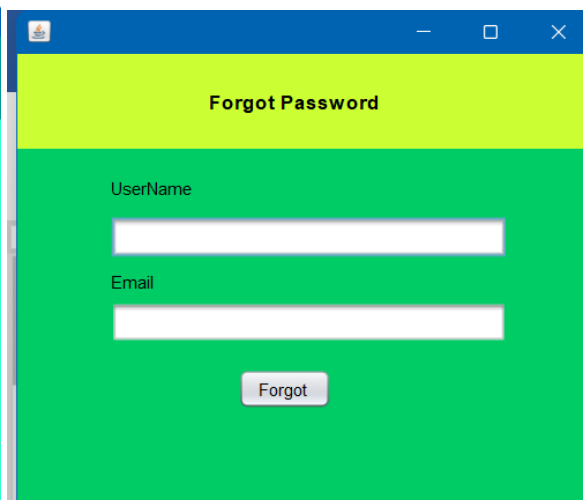
Email

Address

Identity Number

Signup

Login



A screenshot of a web browser window titled "Forgot Password". The window has a blue header bar. Below the header, the title "Forgot Password" is centered in bold black text. The main content area has a green background. It contains two white input fields for "UserName" and "Email". Below the email field is a "Forgot" button.

Forgot Password

UserName

Email

Forgot

Admin Name :

ADMIN PANEL

Notifications

Username: sdfg
Title: asdfg
Days: 5
Date: 2023-04-08

Username: sde
Title: sdfg
Days: 4
Date: 2023-04-08

Username: sde
Title: sdfg
Days: 4
Date: 2023-04-08

Username: sde
Title: sdfg
Days: 4
Date: 2023-04-08

Username: Umer
Title: 1984
Days: 5
Date: 2023-04-09

Username: Umer
Title: The Great Gatsby
Days: 5
Date: 2023-04-09

Username: user
Title: 1984

Delete Book

Delete User

Add Book

Delete

Delete

Add Book

Id

Title

Author

Publisher

Year

Type

Add Book

Add User

Name

Password

Email

Address

Id num

Username

Add User

Export Data

Show Users

Show Books

Logout

Admin Name :

ADMIN PANEL

Notifications

Username: sdfg
Title: asdfg
Days: 5
Date: 2023-04-08

Username: sde
Title: sdfg
Days: 4
Date: 2023-04-08

Username: sde
Title: sdfg
Days: 4
Date: 2023-04-08

Username: sde
Title: sdfg
Days: 4
Date: 2023-04-08

Username: Umer
Title: 1984
Days: 5
Date: 2023-04-09

Username: Umer
Title: The Great Gatsby
Days: 5
Date: 2023-04-09

Username: user
Title: 1984

Delete Book

Delete User

Add Book

Delete

Delete

Add Book

Id

Title

Author

Publisher

Year

Type

Add Book

Add User

Name

Password

Email

Address

Id num

Username

Add User

Export Data

Show Users

Show Books


Logout

Book Id: 1
Title: To Kill a Mockingbird
Author: Harper Lee
Year: 1960
ISBN: 1 01 000000 0
Quantity: Historical fiction

Book Id: 2
Title: 1984
Author: George Orwell
Year: 1949
ISBN: Secker & Warburg
Quantity: political fiction

Book Id: 3
Title: The Great Gatsby
Author: F. Scott Fitzgerald
Year: 1925
ISBN: Charles Scribner's Sons
Quantity: Tragedy

Book Id: 4
Title: Pride and Prejudice
Author: Jane Austen
Year: 1813
ISBN: T. Egerton, Whitehall
Quantity: Romance



LIBRARY MANAGMENT SYSTEM

Uname

Historical fiction
▼

Book

Day

Notifications

Title: 1984, Days: 5
Title: The Great Gatsby, Days: 4
Title: Pride and Prejudice, Days: 55
Total Books Borrowed: 3
Total Days Borrowed: 64

book_id	title	author	publisher	publication_year	Type
1	To Kill a Mockingbird	Harper Lee	J. B. Lippincott & C...	1960	Historical fiction
2	1984	George Orwell	Secker & Warburg	1949	political fiction
3	The Great Gatsby	F. Scott Fitzgerald	Charles Scribner's ...	1925	tragedy
4	Pride and Prejudice	Jane Austen	T. Egerton, Whiteh...	1813	Romance

Database

library

- New
- admins
- books
- borrowedbooks
- users

mysql
performance_schema
phpmyadmin
test

☐ Show all | Number of rows: 25 ▼ | Filter rows:

username	date	title	days
sdfg	2023-04-08	asdfg	5
sde	2023-04-08	sdfg	4
sde	2023-04-08	sdfg	4
sde	2023-04-08	sdfg	4
Umer	2023-04-09	1984	5
Umer	2023-04-09	The Great Gatsby	5
user	2023-04-09	1984	5
Umer	2023-04-27	1984	6

☐ Show all | Number of rows: 25 ▼ | Filter rows:

Test Cases

Test Cases for Login:

1. Test Case Name: Valid Admin Login Input: Username: admin, Password: 123, Radio button: Admin Expected Output: Admin panel should open Test Result: Pass
2. Test Case Name: Valid User Login Input: Username: user, Password: 123, Admin not selected Expected Output: User panel should open Test Result: Pass
3. Test Case Name: Invalid Login Input: Username: test, Password: 123, Radio button: Admin Expected Output: Error message "Invalid Data" should be displayed Test Result: Pass

Test Cases for Delete User:

1. Test Case Name: Valid Delete User Input: User ID: 10 Expected Output: User should be deleted from the database Test Result: Pass
2. Test Case Name: Invalid Delete User Input: User ID: 1000 Expected Output: Error message "User not found" should be displayed Test Result: Pass

Test Cases for Delete Book:

1. Test Case Name: Valid Delete Book Input: Book ID: 20 Expected Output: Book should be deleted from the database Test Result: Pass
2. Test Case Name: Invalid Delete Book Input: Book ID: 2000 Expected Output: Error message "Book not found" should be displayed Test Result: Pass

Test Cases for Forgot Password:

1. Test Case Name: Valid Forgot Password Input: Username: user, Email: user@gmail.com Expected Output: Password should be displayed Test Result: Pass
2. Test Case Name: Invalid Forgot Password Input: Username: test, Email: test@gmail.com Expected Output: Error message "Invalid username or email" should be displayed Test Result: Pass

Test Table:

Test Case Name	Input	Expected Output	Test Result
Valid Admin Login	Username: admin, Password: 123, Radio button: Admin	The admin panel should open	Pass
Valid User Login	Username: user, Password: 123, Radio button: Admin not selected	The user panel should open	Pass
Invalid Login	Username: test, Password: 123, Radio button: Admin	Error message "Invalid Data" should be displayed	Pass
Valid Delete User	User ID: 10	The user should be deleted from the database	Pass
Invalid Delete User	User ID: 1000	Error message "User not found" should be displayed	Pass
Valid Delete Book	Book ID: 20	The book should be deleted from the database	Pass
Invalid Delete Book	Book ID: 2000	Error message "Book not found" should be displayed	Pass
Valid Forgot Password	Username: user, Email: user@gmail.com	Password should be displayed	Pass
Invalid Forgot Password	Username: test, Email: test@gmail.com	Error message "Invalid username or email" should be displayed	Pass

Conclusion

We designed a Java Swing application for a library administration system that allows users to search for books, reserve books, and manage their accounts. We used JDBC to manage data and object-oriented principles like inheritance, polymorphism, and abstraction to create a modular and adaptable system. The software also employs exception handling to control any exceptions that may arise when interacting with the database. Swing components are employed in applications to create an easy-to-use graphical interface.