

A Project Report on
LIBRARY MANAGEMENT SYSTEM



Submitted by:

Theishingkhui R. Shimray

IT A Level

Regd. No: 1210585

Guided by:

Nurzamal Islam

IT Faculty

NIELIT Guwahati City Centre

ABSTRACT

Library is place where all kind of books are available. Library Management system is a desktop-based application. This system contains list of all the books and can be accessed by librarian. This system is mainly two-tier architecture Admin and Librarian. These two modules contain many other sub modules.

The Admin is the one who can see the librarians and their details like address contact number ID password etc. The Admin can add more librarian or delete librarian also as per the requirement.

The Librarian can access the database of the library. Librarians can see the records, number of books available, can add new books to the library, issue books to student, return books from student, see the already issued books etc.

INDEX

S.N CONTENTS

1. INTRODUCTION
2. ANALYSIS
 - 2.1 SYSTEM ANALYSIS
 - 2.2 SYSTEM SECIFICATIONS
3. DESIGN APPROACH
 - 3.1 INTRODUCTION TO DESIGN
 - 3.2 UML DIAGRAMS STUDY
 - 3.3 DATA FLOW DIAGRAMS STUDY
 - 3.4 E-R DIAGRAMS STUDY
4. PROJECT MODULES
5. DATABASE MODELS
6. IMPLEMENTATION
7. OUTPUT SCREENS
8. TESTING
 - 8.1 TEST CASES
9. CONCLUSION
10. FUTURE ENHANCEMENTS
11. BIBILIOGRAPHY

Introduction

Library Management System consists of list of records about the management of the details of the students and the issues going on and also about some books and all. This is a desktop-based application. As the modern organizations are automated and computers are working as per the instructions, it becomes essential for the coordination of human beings, commodity and computers in a modern organization. The administrators and Librarians can communicate with the system through this project, thus facilitating effective implementation and monitoring of various books in the library.

The “Library Management system” has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hard ships faced by this existing system. Moreover, this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

Every organization, whether big or small, has challenges to overcome and managing the information of books, student, librarian, address, members. Every library management system has different needs; therefore, we design exclusive employee management systems that are adapted to your managerial.

The Library Management system is designed & developed for a receipt and issuance of books in the library along with the student’s details. The books received in the library are entered in books entry form and the new student is entered in the student entry form. When the student wants to get the desired book the same is issued on the availability basis to the student.

SYSTEM ANALYSIS OF LIBRARY MANAGEMENT SYSTEM PROJECT

1. Existing System

Various problems of physical system are described below :-

- If one is not very careful then there is a possibility of issuing more than one book to a user.
- There is a possibility of issuing a book to a user, whose membership is not there.
- When a user requests for the a book, one has to physically check for the presence of a book in the library
- Answering management query is a time consuming process.
- Daily keeping a manual record of changes taking place in the library such as book being issued, book being returned etc can become cumbersome if the Library size is bigger.

2. Proposed System

The LIBRARY MANAGEMENT SYSTEM is a software application which avoids more manual hours in taking the book, that need to spend in record keeping and generating reports. Maintaining of user details is complex in manual system in terms of agreements, royalty and activities. This all have to be maintained in ledgers or books. Co-coordinators needs to verify each record for small information also.

- Easy search of book in the online library.
- Avoid the manual work.
- User need not go to the library for Issue any kind of book, he can renewal the book online.

3. Objective of the System

The goal of the system is to bring down the work load with the increased efficiency and to speed up the activities. With this it is very easy to process course fee that is collected time to time from students who are registered and studying at franchisees.

4. Scope

It may help collecting perfect management in details. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of passed year perfectly and vividly. It also helps in current all works relative to library management system. It will be also reduced the cost of collecting the management & collection procedure will go on smoothly.

Our project aims at business process automation, i.e. we have tried to computerize various of Library Management System.

- 1) It satisfy the user requirement
- 2) Be easy to understand by the user and operator
- 3) Have a good user interface
- 4) Be expandable

System Specifications

Hardware Requirements:-

- Pentium-IV(Processor).
- 256 MB Ram
- 512 KB Cache Memory
- Hard disk 10 GB
- Microsoft Compatible 101 or more Key Board

Software Requirements: -

- **Operating System** : Windows 95/98/XP with MS-office
- **Programming language:** Java
- **IDE** : Eclipse
- **Back-End** : MySQL

INTRODUCTION OF LIBRARY MANAGEMENT SYSTEM PROJECT

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

UML Diagrams:

Actor:

A coherent set of roles that users of use cases play when interacting with the use cases.

Use case: A description of sequence of actions, including variants, that a system performs that yields an observable result of value of an actor.

UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

There are various kinds of methods in software design:

They are as follows:

- Use case Diagram
- Sequence Diagram
- Collaboration Diagram
- Activity Diagram
- State chat Diagram

USECASE DIAGRAMS:

Use case diagrams model behavior within a system and helps the developers understand of what the user requires. The stick man represents what's called an actor. Use case diagram can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do.

Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

- The purpose is to show the interactions between the use case and actor.
- To represent the system requirements from user's perspective.
- An actor could be the end-user of the system or an external system.

USECASE DIAGRAM:

A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors. An actor represents a real-world object. Primary Actor – Sender, Secondary Actor Receiver.

SEQUENCE DIAGRAM:

Sequence diagram and collaboration diagram are called INTERACTION DIAGRAMS. An interaction diagram shows an interaction, consisting of set of objects and their relationship including the messages that may be dispatched among them.

A sequence diagram is an introduction that empathizes the time ordering of messages. Graphically a sequence diagram is a table that shows objects arranged along the X-axis and messages ordered in increasing time along the Y-axis

DATA FLOW DIAGRAMS OF LIBRARY MANAGEMENT SYSTEM PROJECT

The DFD takes an input-process-output view of a system i.e. data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software.

Data objects represented by labeled arrows and transformation are represented by circles also called as bubbles. DFD is presented in a hierarchical fashion i.e. the first data flow model represents the system as a whole. Subsequent DFD refine the context diagram (level 0 DFD), providing increasing details with each subsequent level.

The DFD enables the software engineer to develop models of the information domain & functional domain at the same time. As the DFD is refined into greater levels of details, the analyst perform an implicit functional decomposition of the system. At the same time, the DFD refinement results in a corresponding refinement of the data as it moves through the process that embody the applications.

A context-level DFD for the system the primary external entities produce information for use by the system and consume information generated by the system. The labeled arrow represents data objects or object hierarchy.

RULES FOR DFD:

- Fix the scope of the system by means of context diagrams.
- Organize the DFD so that the main sequence of the actions
- Reads left to right and top to bottom.
- Identify all inputs and outputs.
- Identify and label each process internal to the system with Rounded circles.
- A process is required for all the data transformation and Transfers. Therefore, never connect a data store to a data Source or the destinations or another data store with just a Data flow arrow.
- Do not indicate hardware and ignore control information.
- Make sure the names of the processes accurately convey everything the process is done.
- There must not be unnamed process.
- Indicate external sources and destinations of the data, with Squares.
- Number each occurrence of repeated external entities.
- Identify all data flows for each process step, except simple Record retrievals.
- Label data flow on each arrow.
- Use details flow on each arrow.
- Use the details flow arrow to indicate data movements.

E-R Diagrams OF LIBRARY MANAGEMENT SYSTEM PROJECT

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represents data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design For the database designer, the utility of the ER model is:

- it maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
- it is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- In addition, the model can be used as a design plan by the database developer to implement a data model in a specific database management software.

Connectivity and Cardinality

The basic types of connectivity for relations are: one-to-one, one-to-many, and many-to-many. A *one-to-one* (1:1) relationship is when at most one instance of a entity A is associated with one instance of entity B. For example, "employees in the company are each assigned their own office. For each employee there exists a unique office and for each office there exists a unique employee.

A *one-to-many* (1:N) relationships is when for one instance of entity A, there are zero, one, or many instances of entity B, but for one instance of entity B, there is only one instance of entity A. An example of a 1:N relationships is a department has many employees each employee is assigned to one department.

A *many-to-many* (M:N) relationship, sometimes called non-specific, is when for one instance of entity A, there are zero, one, or many instances of entity B and for one instance of entity B there are zero, one, or many instances of entity A. The connectivity of a relationship describes the mapping of associated

ER Notation

There is no standard for representing data objects in ER diagrams. Each modeling methodology uses its own notation. The original notation used by Chen is widely used in academics texts and journals but rarely seen in either CASE tools or publications by non-academics. Today, there are a number of notations used, among the more common are Bachman, crow's foot, and IDEFIX.

All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin. The symbols used for the basic ER constructs are:

- **entities** are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.
- **relationships** are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs
- **attributes**, when included, are listed inside the entity rectangle. Attributes which are identifiers are underlined. Attribute names should be singular nouns.
- **cardinality** of many is represented by a line ending in a crow's foot. If the crow's foot is omitted, the cardinality is one.
- **existence** is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required. Optional existence is shown by placing a circle next to the entity that is optional

PROJECT MODULES

MODULES USED: -

The proposed system consists of the following modules:

Login component

1. Administrator (Head office manager)
2. Librarian

Administrator Component

1. Administrator (Login as Admin, View Librarian, Add Librarian, Delete Librarian, Modify)

Librarian Manager Component

2. Librarian Manager (Manage the database of Library Management System)

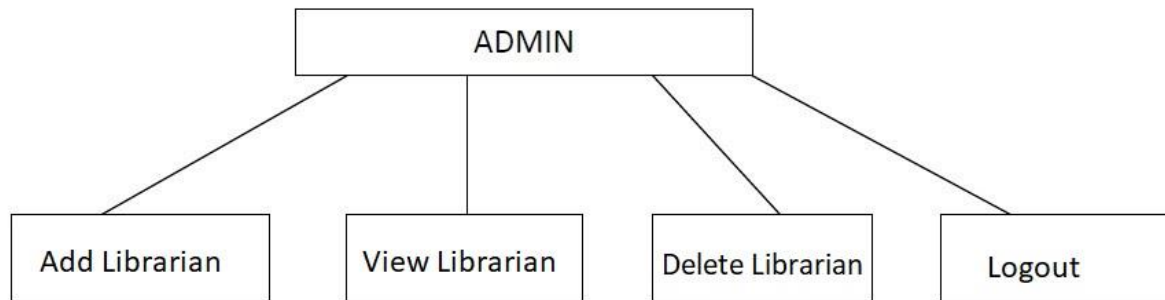
Other Component within Librarian

1. View Book Details
2. Add Books into the Library
3. Delete Books
4. Issue Books to Students
5. View Issued Books
6. Return Books

MODULES DESCRIPTION: -

Admin: Admin is Head of all. Admin can login using User ID and Password provided. Admin can see all the details of librarian like Address, Contact, Phone Number ID, Password etc. Admin can add new librarian or delete an existing librarian.

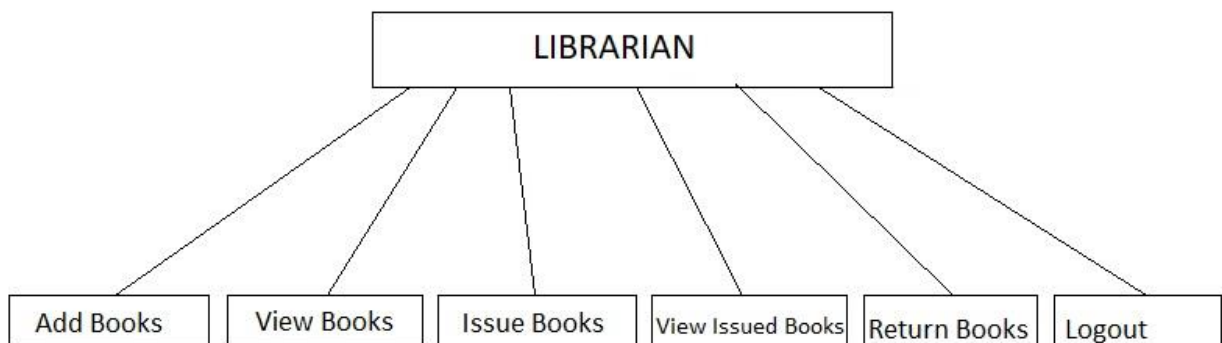
Librarian: Librarian is a person who manages the Library. Librarian has the permission that he



can access the database. There are some tasks which are performed by the Librarian like:

- Addition of a new book.
- Modification of the book.
- Deletion of the book.
- Searching of the book.
- Issue Books
- View Issued Books
- Return Books

Addition of a new book: - This module is accessible by the librarian only. The librarian can add new books into the library mentioning Book Name, Call Number, Author Name, Quantity etc.



Deletion of the book: The Librarian can delete or modify the books from the database

Book Issue module: - This module allows students to give their books to the librarian for issuing that book from the library. A particular student can borrow a maximum of three books only at a time. Each book can be borrowed for a maximum of 14 days.

View Issued Books: This module is also accessible by the librarian. Librarian can see the books issued already to which students and how many numbers of books.

Book Return module: - This module allows students to return books in library. This module is accessible by librarian. Books, which are returned after 14 days from the borrowed date, are charged 1 Rupee per day.

Database Models:

JDBC and accessing the database through applets, and JDBC API via an intermediate server resulted in a new type of database model which is different from the client-server model.

Based on number of intermediate servers through which request should go through named as single tier, two tier and multi-tier architecture.

Single Tier:

In a single tier the server and client are the same in the sense that a client program that needs information and the database are inside the same machine. The advantage with this is the simplicity and portability of the application developed.

Two Tier (Client-Server):

In a two-tier architecture the database resides in one machine(server) and the data can be accessed by any number of machines(clients) in the network. In this type of architecture, a database manager takes control of the database and provides access to clients in a network. This software bundle is also called as the server. Software in different machines, requesting for information are called as clients.

Three tier and N-tier:

The three tier architecture, the database that resides one server, can be accessed by any number of servers, which In turn serve clients in a network .for example, you want to access the database using java applets, the applet running in some other machine, can send requests only to the server from which it is down loaded. For this reason, we will need to have an intermediate server acts as a two-way communication channel also This is, the information or data from the database is passed on to the applet that is recession it. This can extend to make n tiers of servers, each server carrying type of request from clients, however in practice only three tier architecture is more popular.

In this project we have used **Single Tier Architecture Database Model**.

FEASIBILITY STUDY:

Feasibility study is conducted once the problem is clearly understood. Feasibility study is a high-level capsule version of the entire system analysis and design process. The objective is to determine quickly at a minimum expense how to solve a problem. The purpose of feasibility is not to solve the

problem but to determine if the problem is worth solving.

The system has been tested for feasibility in the following points.

1. Technical Feasibility
2. Economic Feasibility
3. Operational Feasibility.

Technical Feasibility

The project entitles "Library Management System" is technically feasibility because of the below mentioned feature. The project was developed in Java which Graphical User Interface.

It provides the high level of reliability, availability and compatibility. All these make Java an appropriate language for this project. Thus, the existing software Java is a powerful language.

Economic Feasibility

The computerized system will help in automate the selection leading the profits and details of the organization. With this software, the machine and manpower utilization are expected to go up by 80-90% approximately. The costs incurred of not creating the system are set to be great, because precious time can be wanted by manually.

Operational Feasibility

In this project, the management will know the details of each book where he may be presented and the data will be maintained as decentralized and if any inquires for that particular contract can be known as per their requirements and necessities.

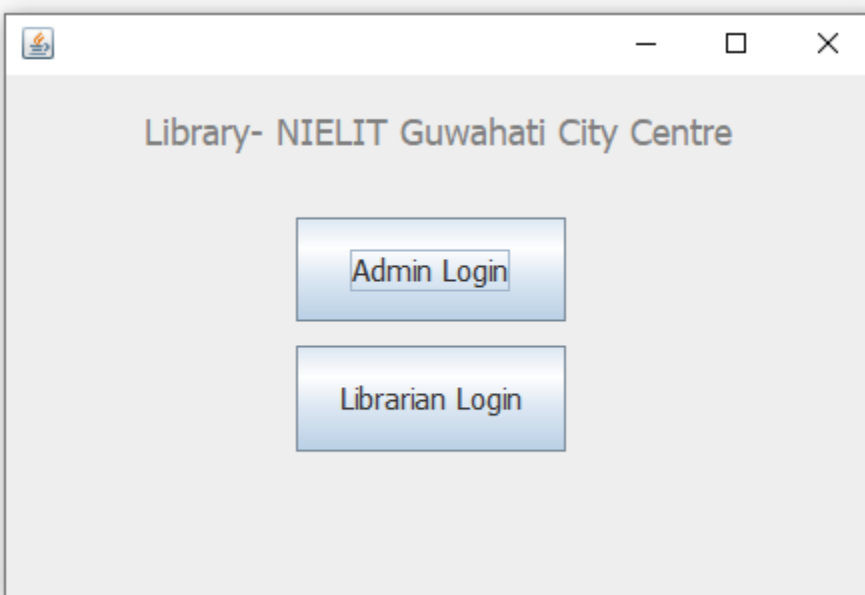
Implementation:

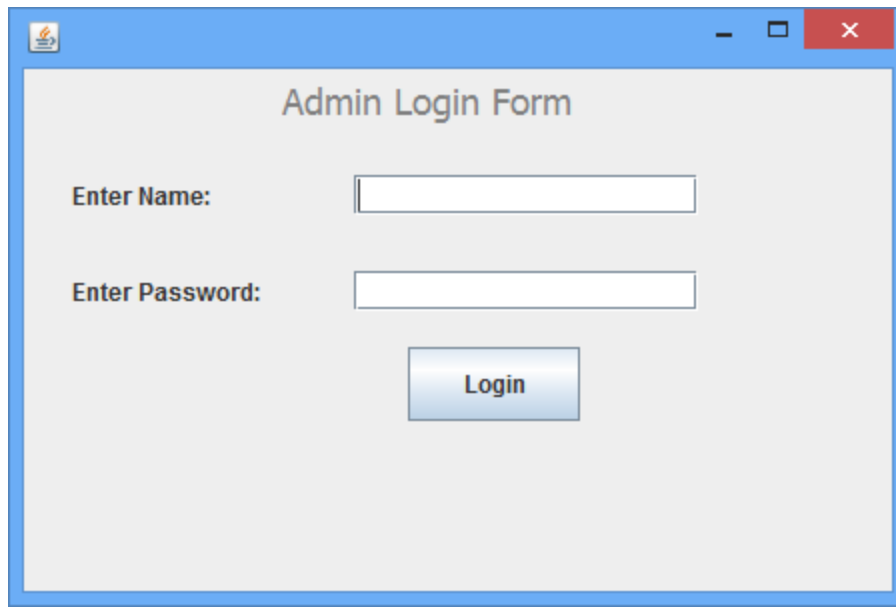
Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively.

It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover and an evaluation of change over methods a part from planning. The more complex the system being implemented, the more involved will be the systems analysis and design effort required just for implementation.

The screenshots of the results are shown below:

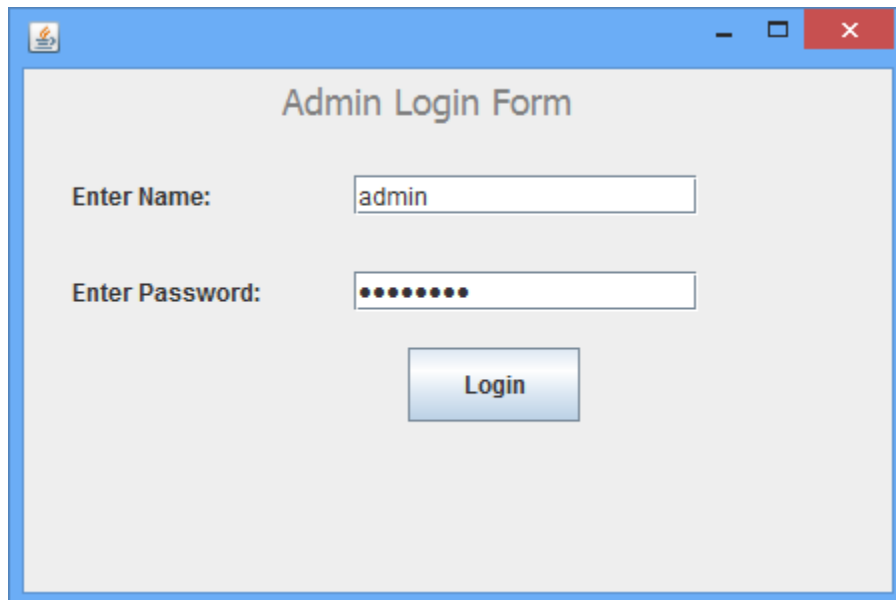
Library Management System Screen Shot



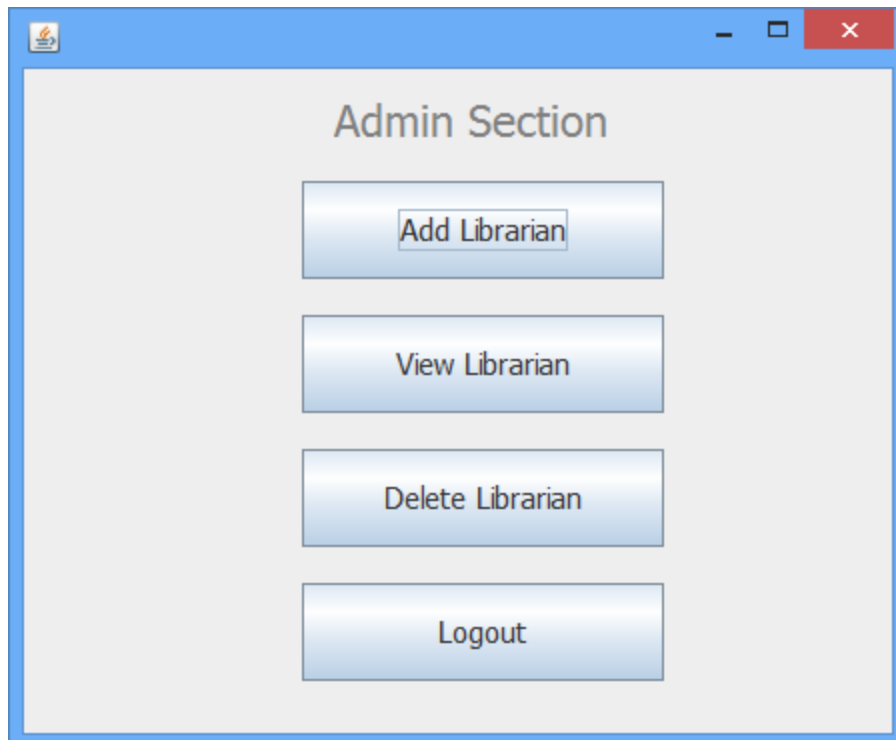


A screenshot of a web application window titled "Admin Login Form". The window has a blue border and standard Windows-style window controls (minimize, maximize, close) in the top right corner. The background is light gray. The form contains two input fields: "Enter Name:" and "Enter Password:". The "Enter Name:" field is empty. The "Enter Password:" field is also empty. Below the input fields is a blue "Login" button.

Write name and password: It must be **admin** for name and **citycentre** for password



A screenshot of the same "Admin Login Form" window, but now with the username "admin" entered in the "Enter Name:" field. The "Enter Password:" field is filled with ten black dots, representing the password "citycentre". The "Login" button remains visible below the fields.



A screenshot of a software window titled "Admin Section". The window has a blue title bar with standard Windows window controls (minimize, maximize, close). The main content area is light gray and contains four blue buttons with white text, arranged vertically in the center. The buttons are labeled "Add Librarian", "View Librarian", "Delete Librarian", and "Logout".

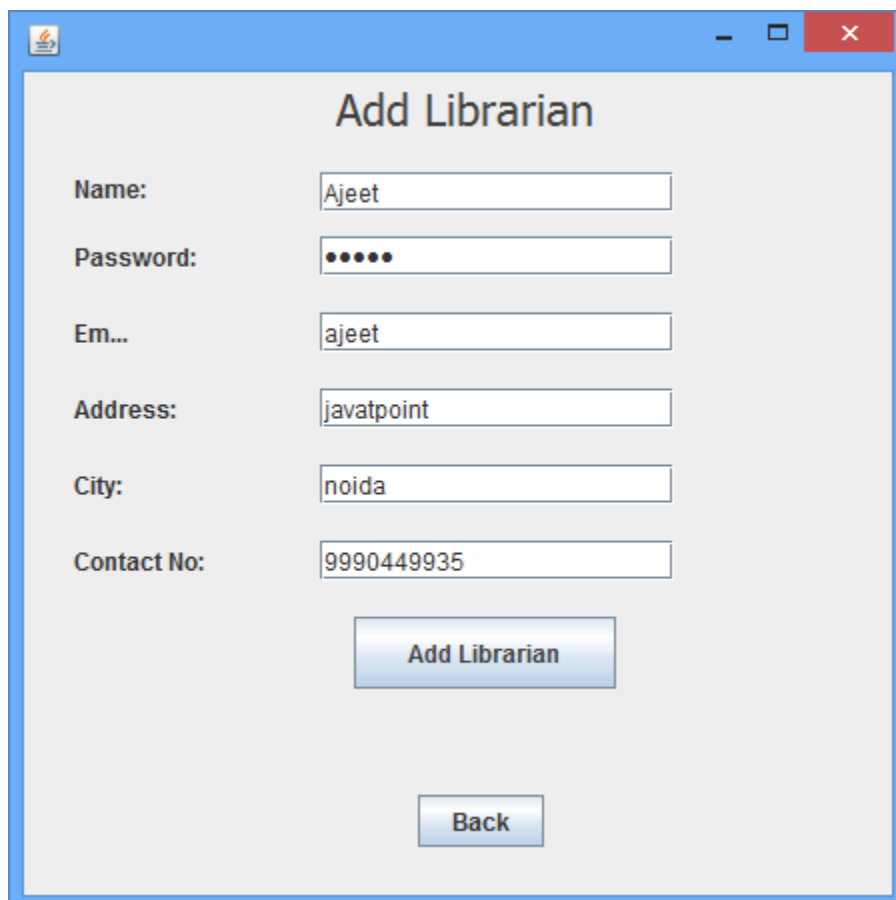
Admin Section

Add Librarian

View Librarian

Delete Librarian

Logout



A screenshot of a software window titled "Add Librarian". The window has a blue title bar with standard Windows window controls. The main content area is light gray and contains a form with several input fields and two buttons. The form fields are labeled "Name:", "Password:", "Em...", "Address:", "City:", and "Contact No:". The input values are "Ajeet", "•••••", "ajeet", "javatpoint", "noida", and "9990449935" respectively. Below the form are two blue buttons with white text: "Add Librarian" and "Back".

Add Librarian

Name: Ajeet

Password: •••••

Em... ajeet

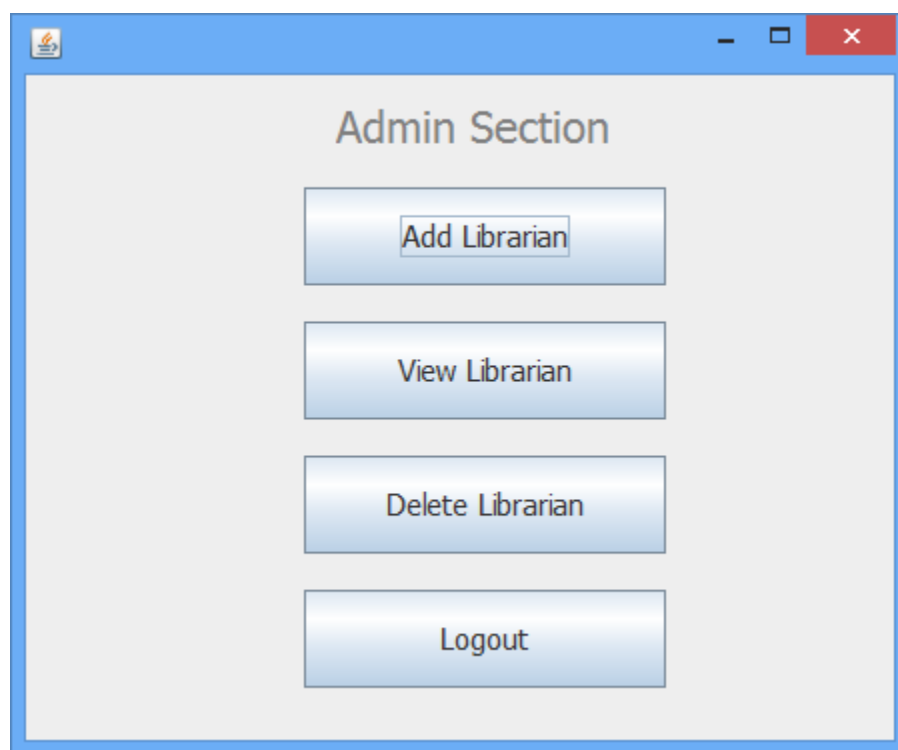
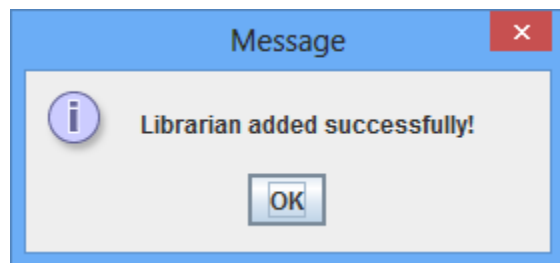
Address: javatpoint

City: noida

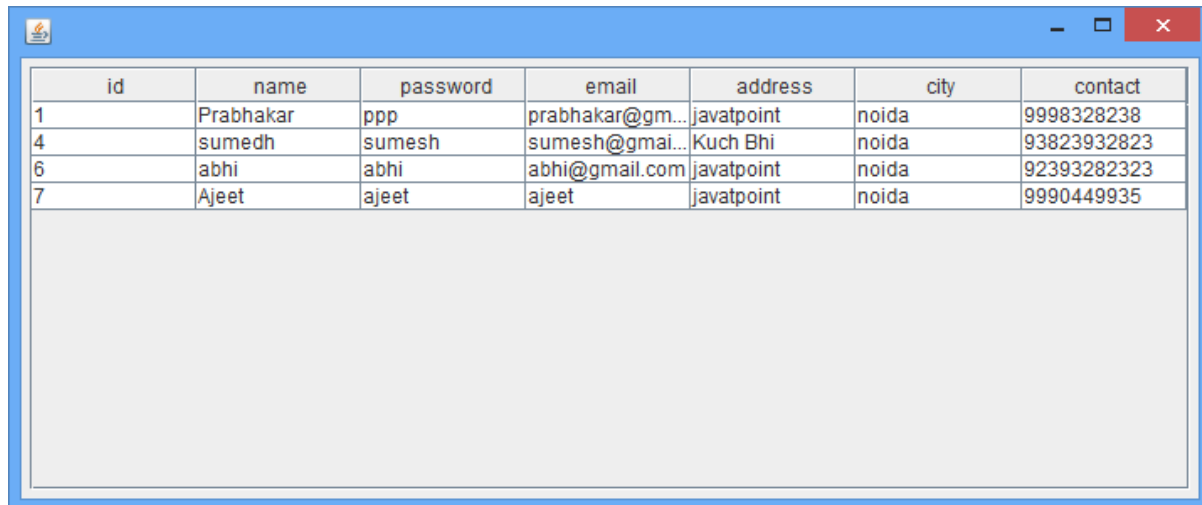
Contact No: 9990449935

Add Librarian

Back



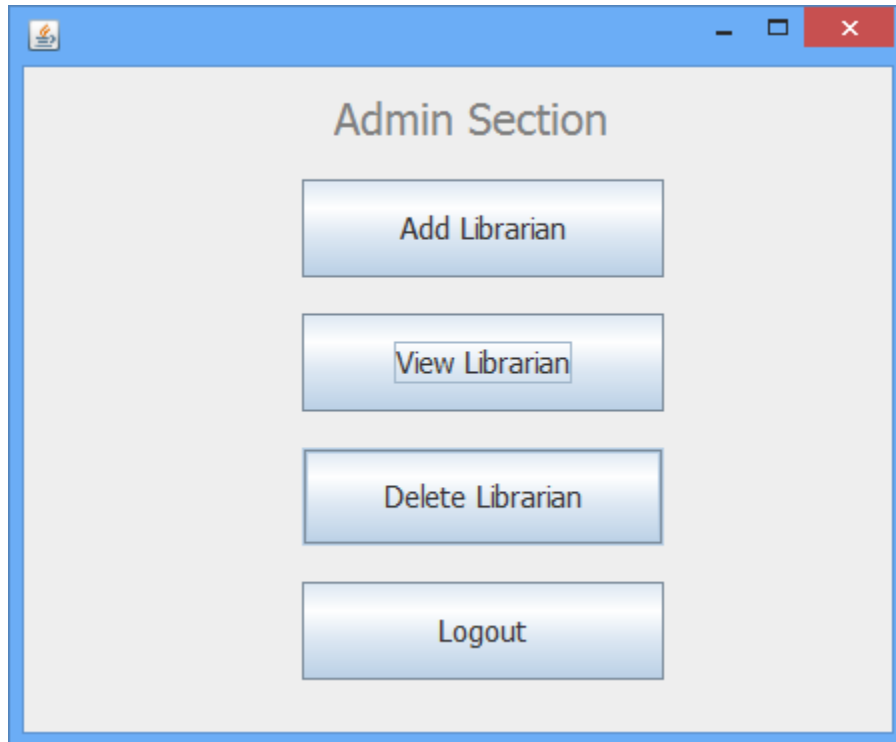
Now click on **view librarian** to see records of librarian.



A screenshot of a web application window with a blue title bar. The window contains a table with 7 columns: id, name, password, email, address, city, and contact. The table has 4 rows of data. Below the table is a large, empty light gray rectangular area.

id	name	password	email	address	city	contact
1	Prabhakar	ppp	prabhakar@gm...	javatpoint	noida	9998328238
4	sumedh	sumesh	sumesh@gmai...	Kuch Bhi	noida	93823932823
6	abhi	abhi	abhi@gmail.com	javatpoint	noida	92393282323
7	Ajeet	ajeet	ajeet	javatpoint	noida	9990449935

Now close the table.



A screenshot of a web application window titled "Admin Section". The window has a blue title bar. The main content area is light gray and contains four blue buttons with white text, arranged vertically: "Add Librarian", "View Librarian", "Delete Librarian", and "Logout".

Now click on **Delete Librarian**.

Enter Id: 4

Delete

Back

Message

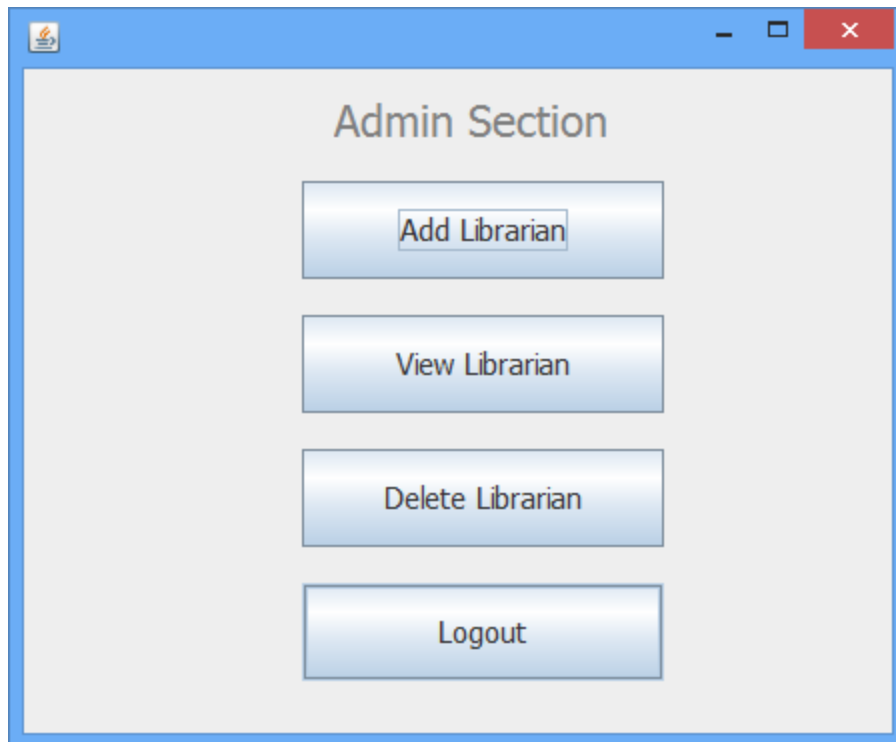
Record deleted successfully!

OK

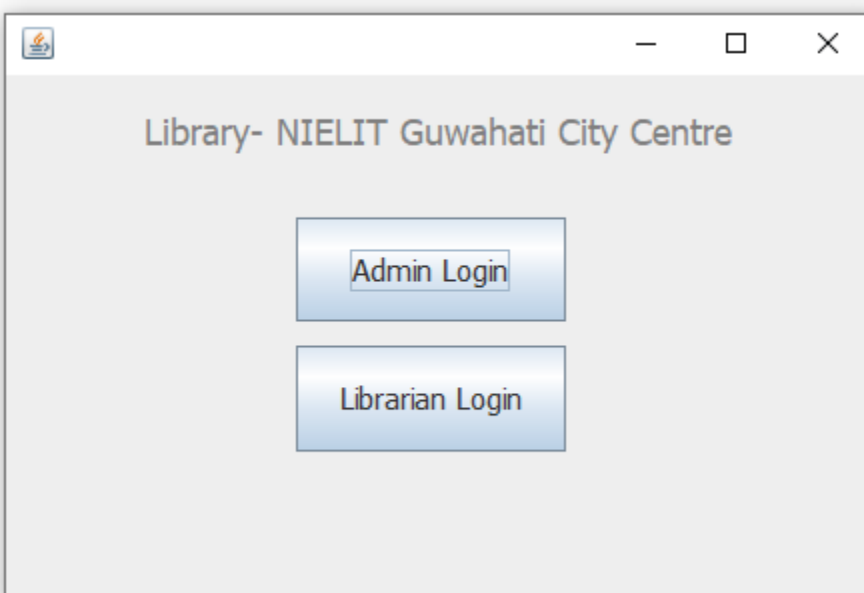
Again click on the **View Librarian** button. You can see librarian is deleted successfully.

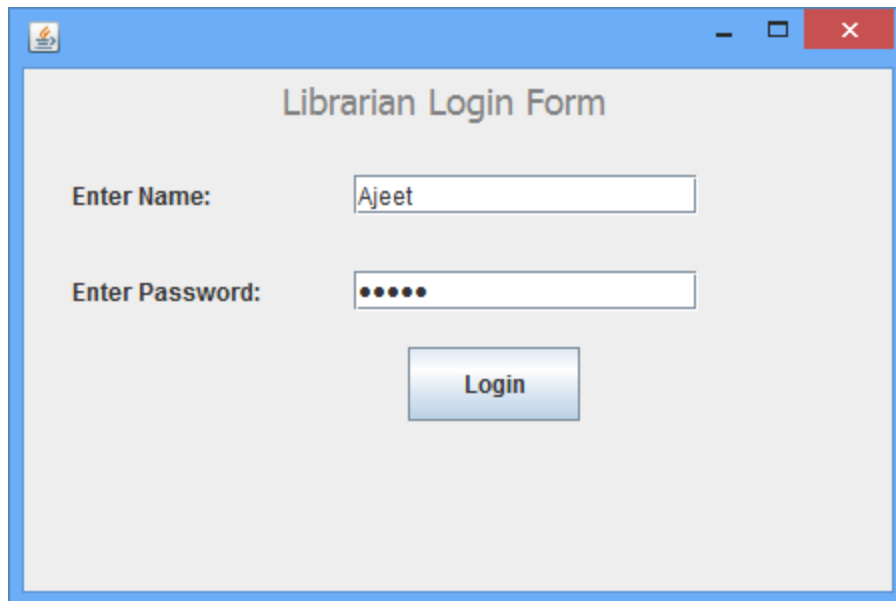
id	name	password	email	address	city	contact
1	Prabhakar	ppp	prabhak...	javatpoint	noida	999832...
6	abhi	abhi	abhi@g...	javatpoint	noida	923932...
7	Ajeet	ajeet	ajeet	javatpoint	noida	999044...

Now close the table.



Now click on the **Logout** button.





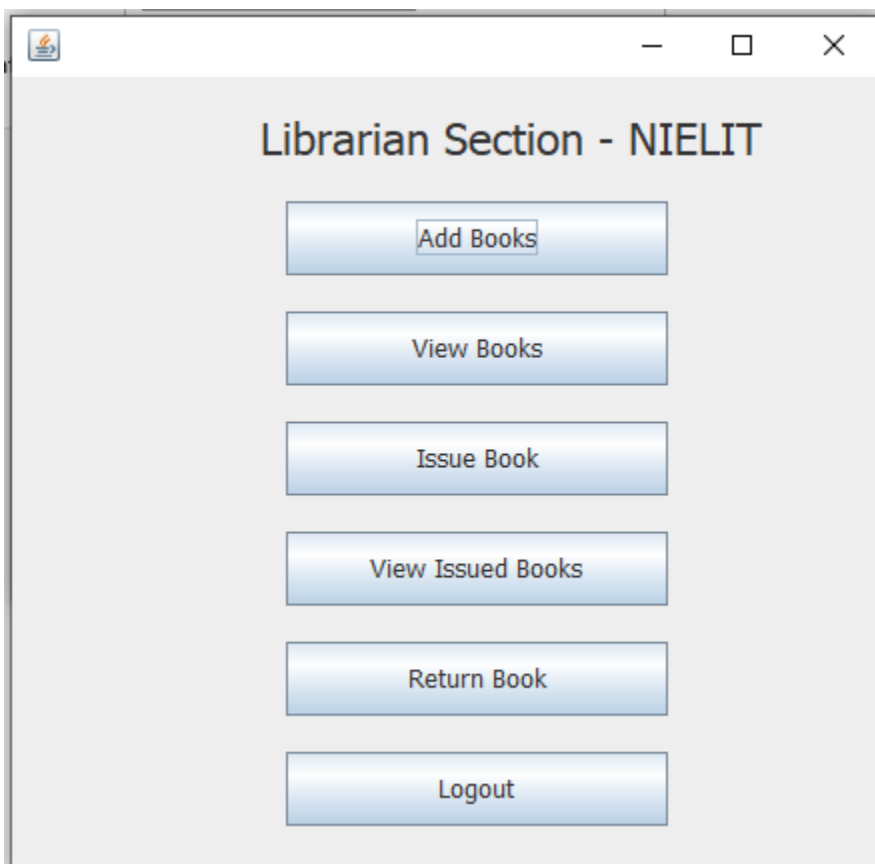
A screenshot of a Java Swing window titled "Librarian Login Form". The window has a blue title bar with standard OS controls. The main area is light gray. It contains two labels: "Enter Name:" and "Enter Password:". The "Enter Name:" label is followed by a text input field containing the text "Ajeet". The "Enter Password:" label is followed by a password input field with five black dots. Below these fields is a blue "Login" button.

Librarian Login Form

Enter Name:

Enter Password:

Login



A screenshot of a Java Swing window titled "Librarian Section - NIELIT". The window has a white title bar with standard OS controls. The main area is light gray. It contains six blue buttons arranged vertically: "Add Books", "View Books", "Issue Book", "View Issued Books", "Return Book", and "Logout".

Librarian Section - NIELIT

Add Books


View Books

Issue Book

View Issued Books

Return Book

Logout

 — □ ×

Add Books

Call No:


Name:

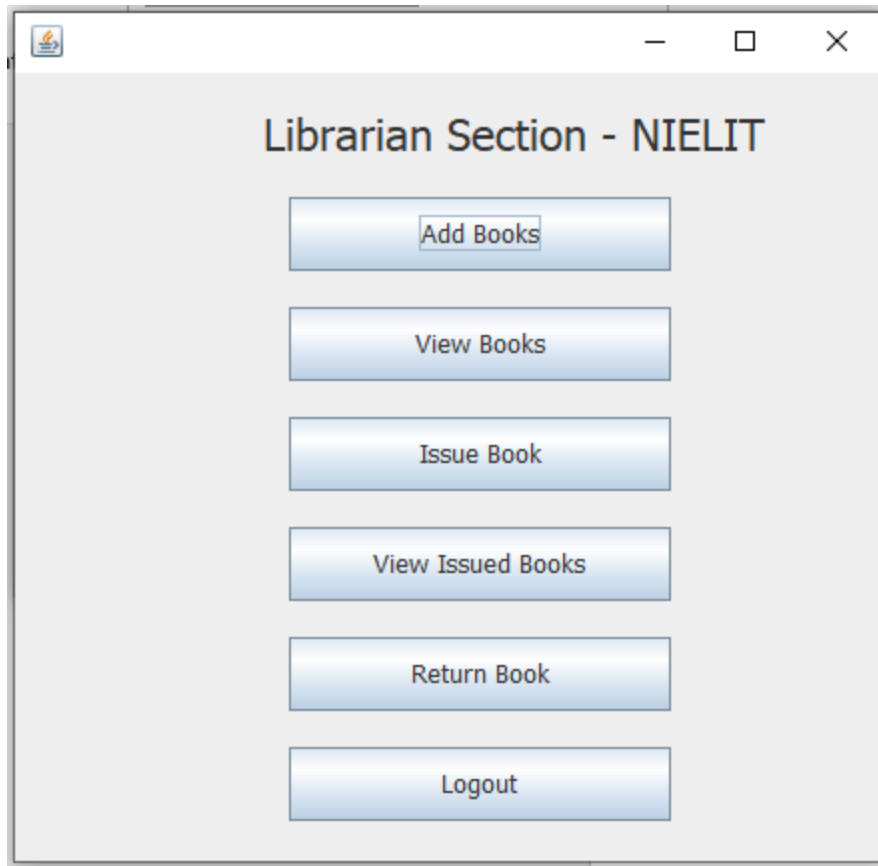
Auth...:

Publisher:

Quantity:

Message ×

 Books added successfully!

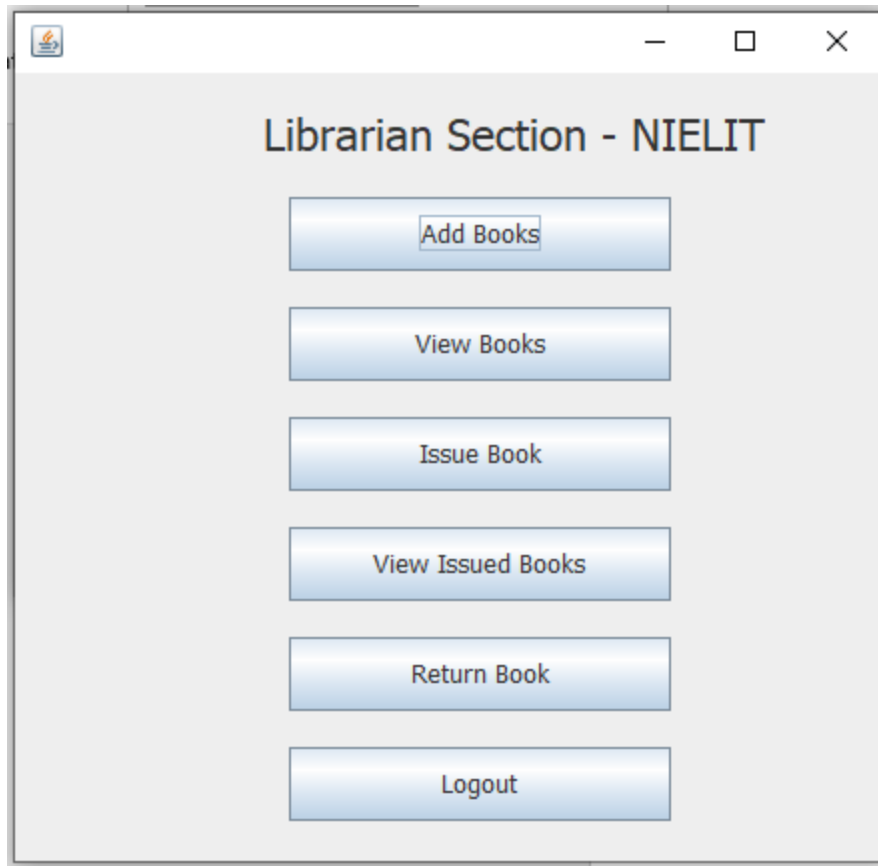


Now click on the **View Books** button. You can see record is added successfully.

A screenshot of a web application window displaying a table of books. The window has a blue title bar and standard Windows window controls. The table has 8 columns: id, callno, name, author, publisher, quantity, issued, and added_date. There are 4 rows of data. Below the table is a large, empty gray rectangular area.

id	callno	name	author	publisher	quantity	issued	added_date
1	A@4	C In Depth	Shrivastav	BPB	2	2	2016-07-20 ...
2	B@1	DBMS	Korth	Pearson	3	0	2016-07-19 ...
3	G@12	Let's see	Yashwant K...	BPB	10	0	2016-07-19 ...
4	K@8	Networking	Forouzan	Tata Mc Gra...	5	0	2016-07-20 ...

Now close the table.



Now click on **Issue Book** button.

Issue Book

Book Callno: K@8

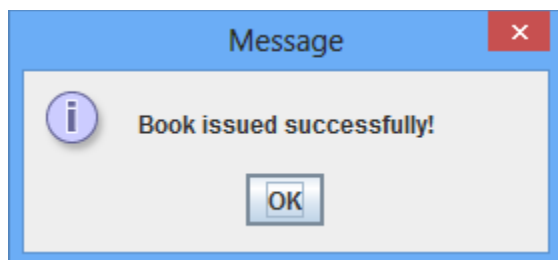
Student Id: 100

Student Name: Vimal


Student Contact: 9990449936

Issue Book Back

Note: Please check Student ID Carefully before issuing book!



Now click on the **view issued books** button.



—

□

×

Librarian Section - NIELIT

Add Books


View Books

Issue Book

View Issued Books

Return Book

Logout

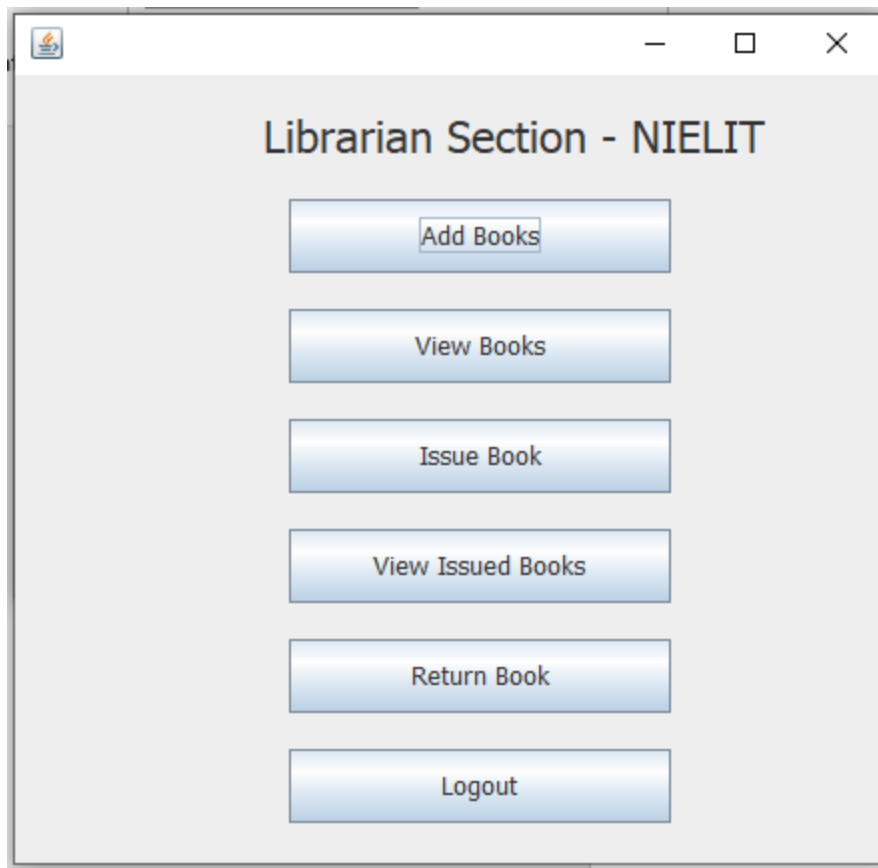


—

□

×

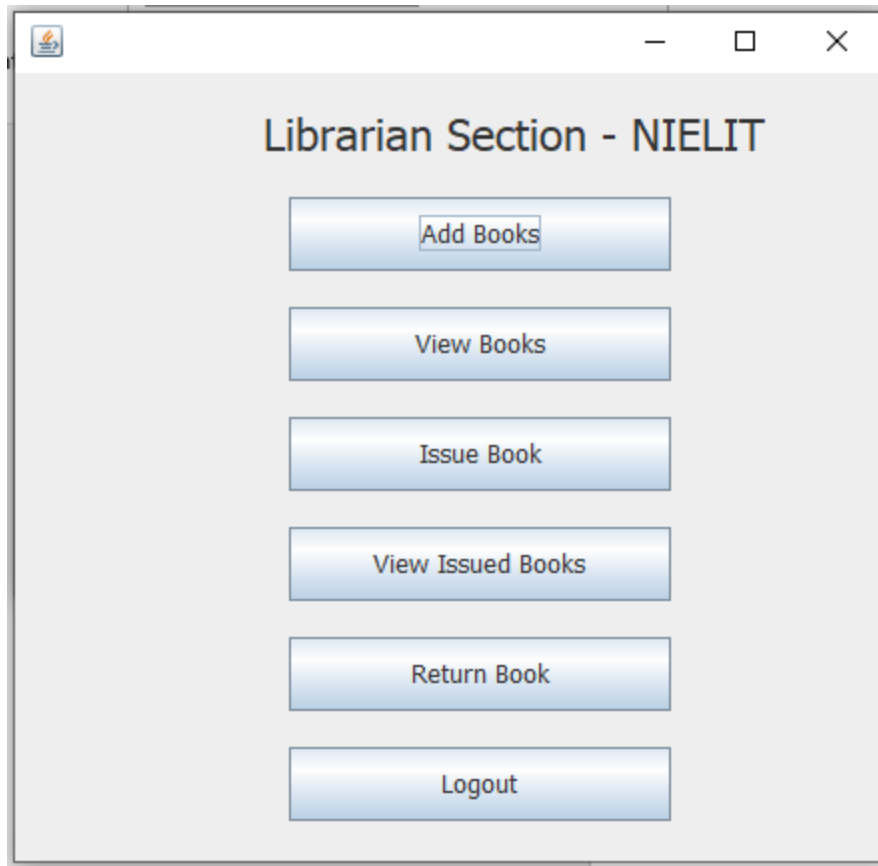
id	bookcallno	studentid	studentname	studentcontact	issueddate
4	A@4	23	kk	932992932	2016-07-20 00:13...
6	A@4	335	Sumedh	95676565756	2016-07-20 00:14...
7	A@4	87	abhishek	9329882382	2016-07-20 00:16...
8	K@8	100	Vimal	9990449936	2016-07-20 01:38...



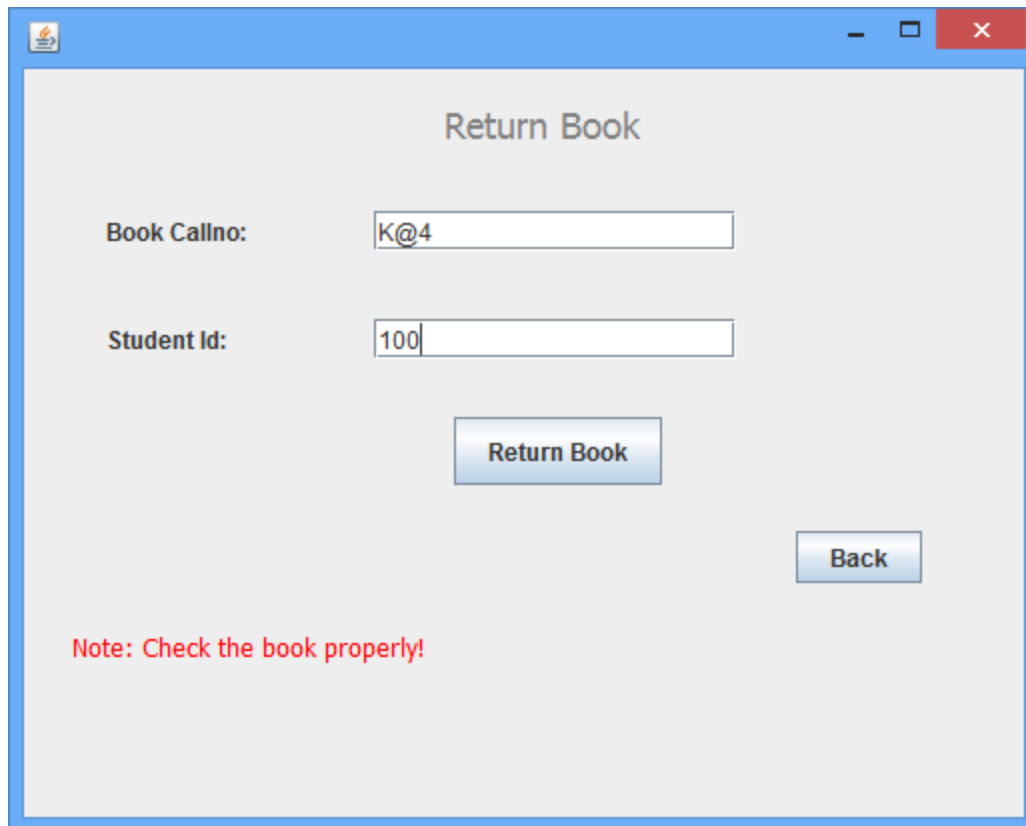
Click on the **View Books** and see **quantity is decremented** and **issued is incremented**.

The screenshot shows a table with 8 columns: id, callno, name, author, publisher, quantity, issued, and added_date. The table contains 4 rows of data. Below the table is a large empty rectangular area.

id	callno	name	author	publisher	quantity	issued	added_date
1	A@4	C In Depth	Shrivastav	BPB	2	2	2016-07-20 ...
2	B@1	DBMS	Korth	Pearson	3	0	2016-07-19 ...
3	G@12	Let's see	Yashwant Ka...	BPB	10	0	2016-07-19 ...
4	K@8	Networking	Forouzan	Tata Mc Gra...	4	1	2016-07-20 ...



Now click on the **Return Book** button.



A Java Swing dialog box titled "Return Book" with a blue title bar and standard window controls. It contains two text input fields: "Book Callno:" with the value "K@4" and "Student Id:" with the value "100". Below the fields are two buttons: "Return Book" and "Back". At the bottom, a red text label reads "Note: Check the book properly!".

Return Book

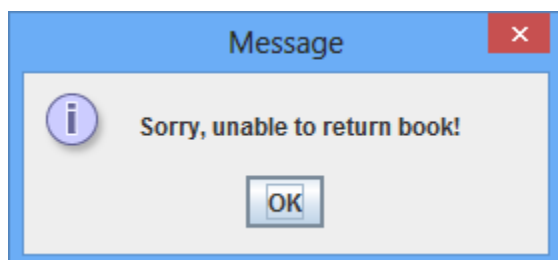
Book Callno: K@4

Student Id: 100

Return Book

Back

Note: Check the book properly!



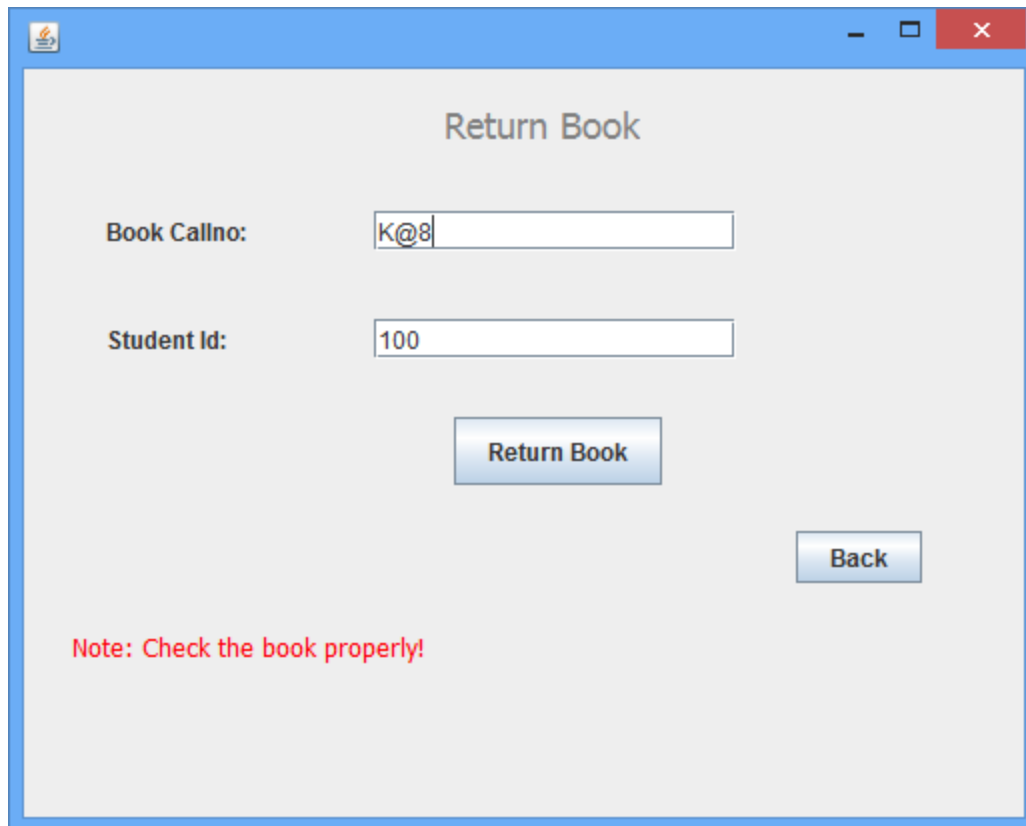
A Java Swing message dialog box titled "Message" with a blue title bar and a close button. It features an information icon (i) on the left and the text "Sorry, unable to return book!". An "OK" button is centered at the bottom.

Message

Sorry, unable to return book!

OK

Write the correct Call no and Student Id:



A dialog box titled "Return Book" with a blue border and standard Windows window controls. It contains two input fields: "Book Callno:" with the value "K@8" and "Student Id:" with the value "100". Below these fields are two buttons: "Return Book" and "Back". At the bottom left, there is a red text note: "Note: Check the book properly!"

Return Book

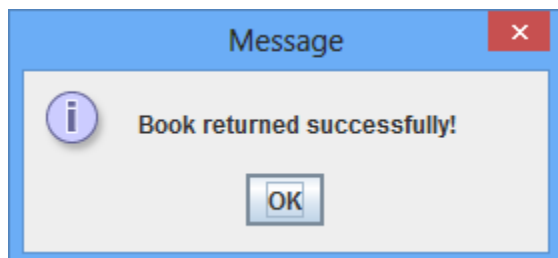
Book Callno: K@8

Student Id: 100

Return Book

Back

Note: Check the book properly!



A message dialog box titled "Message" with a blue border and a close button. It features an information icon (i) on the left and the text "Book returned successfully!". At the bottom center is an "OK" button.

Message


Book returned successfully!

OK

Again click on the **view issued book** button, record is deleted.

id	bookcallno	studentid	studentname	studentcontact	issueddate
4	A@4	23	kk	932992932	2016-07-20 00:1...
6	A@4	335	Sumedh	95676565756	2016-07-20 00:1...
7	A@4	87	abhishek	9329882382	2016-07-20 00:1...

Now close the table.


— □ ×

Librarian Section - NIELIT

Add Books
View Books
Issue Book
View Issued Books
Return Book
Logout

Click on the **View Books** button. You can see that **quantity is incremented** and **issued is decremented**.

id	callno	name	author	publisher	quantity	issued	added_date
1	A@4	C In Depth	Shrivastav	BPB	2	2	2016-07-20...
2	B@1	DBMS	Korth	Pearson	3	0	2016-07-19...
3	G@12	Let's see	Yashwant K...	BPB	10	0	2016-07-19...
4	K@8	Networking	Forouzan	Tata Mc Gra...	5	0	2016-07-20...

TESTING:

The testing phase is an important part of software development. It is the pauperized system will help in automate process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

Software testing is carried out in three steps:

1. The first includes unit testing, where in each module is tested to provide its correctness, validity and also determine any missing operations and to verify whether the objectives have been met. Errors are noted down and corrected immediately. Unit testing is the important and major part of the project. So errors are rectified easily in particular module and program clarity is increased. In this project entire system is divided into several modules and is developed individually. So unit testing is conducted to individual modules.

2. The second step includes Integration testing. It need not be the case, the software whose modules when run individually and showing perfect results, will also show perfect results when run as a whole. The individual modules are clipped under this major module and tested again and verified the results. This is due to poor interfacing, which may results in data being lost across an interface. A module can have inadvertent, adverse effect on any other or on the global data structures, causing serious problems.

3. The final step involves validation and testing which determines which the software functions as the user expected. Here also some modifications were. In the completion of the project it is satisfied fully by the end user.

The primary objective for test case design is to derive a set of tests. To accomplish this objective two different categories of test case design techniques are used. They are

- White box testing.
- Black box testing.

White-box testing:

White box testing focus on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Block-box testing:

Black box testing is designed to validate functional requirements without regard to the internal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides through test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are the errors falling in this category.

Test Cases:

Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Using White-Box testing methods, the software engineer can drive test cases that

- Guarantee that logical decisions on their true and false sides.
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.
- Exercise internal data structure to assure their validity.

The test case specification for system testing has to be submitted for review before system testing commences.

CONCLUSION:

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project.

- Library Management System of the entire system improves the efficiency.
- It provides a friendly graphical user interface which proves to be better when compared to the existing system.
- It gives appropriate access to the authorized users depending on their permissions.
- It effectively overcomes the delay in communications.
- Updating of information becomes so easier.
- System security, data security and reliability are the striking features.
- The System has adequate scope for modification in future if it is necessary.

FUTURE ENHANCEMENTS:

Well I have worked hard in order to present an improved application than the existing one's regarding the information about the various activities. Still, I found out that the project can be done in a better way. The next enhancement can be done using distributed system, where student can login and search for books and request the librarian to issue the particular book for him/her.

BIBLIOGRAPHY

The following books were referred during the analysis and execution phase of the project

- 1) Head First Java, 2nd Edition
- 2) Effective Java (2nd Edition)

WEBSITES:

www.guru99.com

www.tutorialspoint.com