

# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Santhibastawad Road, Machhe  
Belagavi - 590018, Karnataka, India



## DBMS LABORATORY WITH MINI PROJECT (18CSL58) REPORT ON

### “Library Management System”

Submitted in the partial fulfillment of the requirements for the award of the degree of

## BACHELOR OF ENGINEERING IN INFORMATION SCIENCE AND ENGINEERING

For the Academic Year 2022-2023

Submitted by

Namratha Prakash  
Neha Subrahmanya Hegde

1JS20IS054  
1JS20IS056

Under the Guidance of

Dr. Sowmya KN  
Associate Professor  
Dept. of ISE, JSSATEB



2022-2023

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING  
JSS ACADEMY OF TECHNICAL EDUCATION

JSS Campus, Dr. Vishnuvardhan Road, Bengaluru-560060

**JSS MAHAVIDYAPEETHA, MYSURU**  
**JSS ACADEMY OF TECHNICAL EDUCATION**  
JSS Campus, Dr.Vishnuvardhan Road, Bengaluru-560060

**DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING**



**CERTIFICATE**

This is to certify that DBMS LABORATORY WITH MINI PROJECT (18CSL58) Report entitled “**Library Management System**” is a Bonafide work carried out by **Namratha Prakash [1JS20IS054]**, **Neha Subrahmanya Hegde [1JS20IS056]** in partial fulfillment for the award of degree of Bachelor of Engineering in Information Science and Engineering of Visvesvaraya Technological University Belagavi during the year 2022- 2023.

---

**Signature of the Guide**

**Dr. Sowmya K N**  
Associate Professor  
Dept. of ISE  
JSSATEB

---

**Signature of the HOD**

**Dr. Rekha PM**  
Professor & HOD  
Dept. of ISE,  
JSSATEB

# TABLE OF CONTENTS

ACKNOWLEDGEMENT	1
ABSTRACT	2
CHAPTER 1	3
INTRODUCTION	3
INTRODUCTION	3
OBJECTIVES	3
ORGANIZATION OF THE REPORT	4
CHAPTER 2	5
LITERATURE SURVEY	5
INTRODUCTION	5
NORMALIZATION	10
CHAPTER 3	12
REQUIREMENT SPECIFICATIONS	12
FUNCTIONAL REQUIREMENTS	12
USECASE DIAGRAM	12
NON-FUNCTIONAL REQUIREMENTS	13
SOFTWARE SPECIFICATION	14
HARDWARE SPECIFICATION	14
USER CHARACTERISTICS	14
CHAPTER 4	15
SYSTEM DESIGN	15
INTRODUCTION TO SYSTEM DESIGN	15
ATTRIBUTES	15
SCHEMA DIAGRAM	17
ER DIAGRAM	18
CHAPTER 5	20
WIREFRAMES	20
INTRODUCTION TO WIREFRAMES	20
CHAPTER 6	31
PROJECT IMPLEMENTATION	31
INTRODUCTION	31
CREATING TABLES	31
QUERIES	38
CHAPTER 7	41

SYSTEM TESTING	41
INTRODUCTION	41
TYPES OF TESTING	41
CHAPTER 8	45
RESULTS AND DISCUSSIONS	45
CHAPTER 9	60
CONCLUSION AND FUTURE ENHANCEMENTS	60
CONCLUSION	60
FUTURE ENHANCEMENT	60
CHAPTER 10	61
REFERENCES	61
BOOK REFERENCES	61
WEB REFERENCES	61

## ACKNOWLEDGEMENT

---

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible. So with gratitude, we acknowledge all those whose guidance and encouragement crowned our efforts with success.

First and foremost, we would like to thank his **Holiness Jagadguru Sri Shivarathri Deshikendra Mahaswamiji** and **Dr. Bhimasen Soragaon**, Principal, JSSATE, Bangalore for providing an opportunity to carry out the Project Work as a part of our curriculum in the partial fulfilment of the degree course.

We express our sincere gratitude for our beloved Head of the department, **Dr. Rekha P.M.**, for her co-operation and encouragement at all the moments of our approach.

It is our pleasant duty to place on record our deepest sense of gratitude to our respected guide, **Dr. Sowmya K N**, Associate Professor for the constant encouragement, valuable help and assistance in every possible way.

We would like to thank all ISE Department Teachers, our instructor at laboratory **Mrs. Sowmya G** ma'am and other non-teaching staff for providing us with their valuable guidance and for being there at all stages of our work.

**Namratha Prakash[1JS20IS054]**

**Neha Subrahmanya Hegde[1JS20IS056]**

## ABSTRACT

---

The Library Management System is an intranet-based application which can be accessed throughout the campus. This system can view available books, find out who is having the particular books, who is has returned books, the date and the time of borrow and return, books out of stock etc. This is an integrated system that contains both student component and integrated component. The purpose of library management system is to automate the existing manual system by the help of computerized equipment and full fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required hardware and software are available and easy to work with.

Library management system, as described above, can lead to error free, secure and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help the university in better utilization of resources and keep a proper track on transaction of books by maintaining the computerized records without redundant entries.

The project has many features like facility of user/student and facility of librarian/teachers login. The librarian or admin can monitor the whole system. It provides facility for students to see the list of books issued, issue date and return date after logging in whereas, the librarian after logging into his account that is admin can generate various reports, add or delete students, course and book category. Overall this project of ours is being developed to help the students as well as staff of library to maintain the library in best way possible and also reduce the human efforts.

# CHAPTER 1

## INTRODUCTION

---

### INTRODUCTION

The project titled Library Management System is Library Management software for controlling and monitoring the transaction in a library. The project “Library Management System” is developed in php, which mainly focuses on basic operations in a library like adding new book, adding new student searching books, borrow and return books.

Library is regarded as the brain of any institute. Many institutes understand the importance of the library to the growth of the institute and their esteem users (students). Library Management System supports the general requirement of the library like acquisitions, cataloguing and circulation.

### OBJECTIVES

The Objectives of **LIBRARY MANAGEMENT SYSTEM** are:

1. The main objective of our project is to manage the details of Books, Students, Admins, Issues, Transactions etc. by providing a computerized management system for a library which helps the user to manage the records and operations more effectively. It tracks all the information of Librarian, Students, Books etc. In case any faculty wants to *retrieve information* of previously conducted research, this mini-project will provide an appropriate solution.
2. This project has the capability to issue and return books, has separate accounts for admin and student with customized profile photo of the user, tracks all the books issued, keeps the database up-to-date, stores all books and user data in a proper manner with search facility, provides information about the book and day-to-day transaction report.

## ORGANIZATION OF THE REPORT

**Chapter 1** provides the information about the project, and the objectives. In **chapter 2**, we discuss the literature survey and the normalization. In **Chapter 3**, we discuss the software and hardware requirements to run the above applications. **Chapter 4** gives the idea of the system design. **Chapter 5** gives a clear picture about the project and its actual implementation. In **Chapter 6**, we discuss the system testing. **Chapter 7** discusses the results and discussions of the program. **Chapter 8** concludes by giving the direction for future enhancement and the **Chapter 9** includes the references.



## CHAPTER 2

# LITERATURE SURVEY

---

### INTRODUCTION

A database management system (DBMS) refers to the technology for creating and managing databases. Basically, DBMS is a software tool to organize (create, retrieve, update and manage) data in a database. The main aim of a DBMS is to supply a way to store up and retrieve database information that is both convenient and efficient. By data, we mean known facts that can be recorded and that have embedded meaning. Normally people use software such as DBASE IV or V, Microsoft ACCESS, or EXCEL to store data in the form of a database. A datum is a unit of data. Meaningful data combined to form information. Hence, information is interpreted data – data provided with semantics. MS. ACCESS is one of the most common examples of database management software. The name indicates what the database is. Database is one of the important components for many applications and is used for storing a series of data in a single set. In other words, it is a group / package of information that is put in order so that it can be easily access, manage and update.

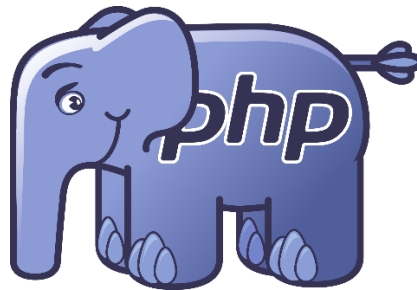
### MySQL

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. It is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons –It is released under an open-source license. So, you have nothing to pay to use it, it is a very powerful program in its own right and handles a large subset of the functionality of the most expensive and powerful database packages. MySQL uses a standard form of the well-known SQL data language. It works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc and works very quickly and works well even with large data sets.



## PHP

PHP Hypertext Preprocessor is a scripting language that helps people make web pages more interactive by allowing them to do more things. PHP code is run on the web server. A website programmed with PHP can have pages that are password protected. A website with no programming cannot do this without other complex things. Standard PHP file extensions are: .php .php3 or .phtml, but a web server can be set up to use any extension. Its structure was influenced by many languages like C, Perl, Java, C++, and even Python. It is considered to be free software by the Free Software Foundation.



## CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colours and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting. The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.



## HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as `<img />` and `<input />` directly introduce content into the page. Other tags such as `<p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.



## JAVASCRIPT

JavaScript, abbreviated as JS, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behavior, and all major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). JavaScript engines were originally used only in web browsers, but they are now embedded in some servers, usually via Node.js. They are also embedded in a variety of applications created with frameworks such as Electron and Cordova.



## Windows 11

**Windows 11** is the latest major release of Microsoft's Windows NT operating system, released in October 2021. It is a free upgrade to its predecessor, Windows 10 (2015), and is available for any Windows 10 devices that meet the new Windows 11 system requirements. Windows 11 features major changes to the Windows shell influenced by the cancelled Windows

10X, including a redesigned Start menu, the replacement of its "live tiles" with a separate "Widgets" panel on the taskbar, the ability to create tiled sets of windows that can be minimized and restored from the taskbar as a group, and new gaming technologies inherited from Xbox Series X and Series S such as Auto HDR and Direct Storage on compatible hardware. Internet Explorer (IE) has been replaced by the Chromium-based Microsoft Edge as the default web browser. Citing security considerations, the system requirements for Windows 11 were increased over Windows 10. Microsoft only officially supports the operating system on devices using an eighth-generation Intel Core CPU or newer (with some minor exceptions), a second-generation AMD Ryzen CPU or newer, or a Qualcomm Snapdragon 850 ARM system-on-chip or newer, with UEFI secure boot and Trusted Platform Module (TPM) 2.0 supported and enabled (although Microsoft may provide exceptions to the TPM 2.0 requirement for OEMs). While the OS can be installed on unsupported processors, Microsoft does not guarantee the availability of updates. Windows 11 removed support for 32-bit x86 CPUs and devices that use BIOS firmware.



## Mac OS X

Mac OS X was originally presented as the tenth major version of Apple's operating system for Macintosh computers; until 2020, versions of macOS retained the major version number "10". The letter "X" in Mac OS X's name refers to the number 10, a Roman numeral, and Apple has stated that it should be pronounced "ten" in this context. However, it is also commonly pronounced like the letter "X". Previous Macintosh operating systems (versions of the classic Mac OS) were named using Arabic numerals, as with Mac OS 8 and Mac OS 9.



## NORMALIZATION

Database normalization is a database schema design technique, by which an existing schema is modified to minimize redundancy and dependency of data. Normalization splits a large table into smaller tables and defines relationships between them to increase the clarity in organizing data. Below are the database normalization types

### First Normal Form (1NF)

- First normal form (1NF) deals with the ‘shape’ of the record type
- A relation is in 1NF if, and only if, it contains no repeating attributes or groups of attributes.
- Example: The Student table with the repeating group is not in 1NF

The domains of attributes in our project includes only atomic (simple, indivisible) values. Hence the database of Library Management System is in First Normal Form.

### Second Normal Form (2NF)

A relation is in 2NF if, and only if, it is in 1NF and every non-key attribute is fully functionally dependent on the whole key.

The Library Management System is in Second Normal Form as every nonprime attribute in every relation is fully functionally dependent on the primary key of that relation.

Example:

In table *borrow*

Primary keys  $\rightarrow$  (id, student\_id, book\_id)

Non prime attributes  $\rightarrow$  (status, date\_borrow)

Here, the non-prime attributes are fully functionally dependent on the primary keys.

### Third Normal Form (3NF)

- A relation is in 3NF if, and only if, it is in 2NF and there are no transitive functional dependencies.
- Transitive functional dependencies arise:
  - when one non-key attribute is functionally dependent on another non-key attribute
  - FD: non-key attribute  $\rightarrow$  non-key attribute and when there is redundancy in the database

The database of Library Management System is in Third normal form as every non key attribute is non transitively dependent on the primary key.

### Boyce-Codd Normal Form (BCNF)

- When a relation has more than one candidate key, anomalies may result even though the relation is in 3NF.
- 3NF does not deal satisfactorily with the case of a relation with overlapping candidate keys
- i.e., composite candidate keys with at least one attribute in common.
- BCNF is based on the concept of a determinant.

The database of Library Management System is not in Boyce-Codd Normal Form as every determinant is a not candidate key.

Example:

In table *borrow*

$(id, student\_id, book\_id) \rightarrow (status, date\_borrow)$

$date\_borrow \rightarrow book\_id$

Here, the *date\_borrow* can't be determined if *student\_id* is not given.

### Fourth Normal Form (4NF)

It is a normal form used in database normalization Introduced by Ronald Fagin in 1977, 4NF is the next level of normalization after Boyce–Codd normal form (BCNF). Whereas the second, third, and Boyce–Codd normal forms are concerned with functional dependencies, 4NF is concerned with a more general type of dependency known as a multivalued dependency.

The database of Library Management System is not in Fourth Normal Form as it is not in BCNF.

### Fifth Normal Form (5NF)

It is also known as project-join normal form (PJ/NF) is a level of database normalization designed to reduce redundancy in relational databases recording multi-valued facts by isolating semantically related multiple relationships. A table is said to be in the 5NF if and only if every non trivial join dependency in that table is implied by the candidate keys.

The database of Library Management System is not in Fourth Normal Form as it is not in Fourth Normal Form.

## CHAPTER 3

# REQUIREMENT SPECIFICATIONS

### FUNCTIONAL REQUIREMENTS

- Allow the librarian to add and remove new members.
- Allow the librarian to update the profile.
- Allow the librarian to add course and book category.
- Allow the student to search for books based on title, publication date, author, etc.
- Student can view borrowed books, returned books and book list.
- Librarian can add and manage the books.

A more detailed list of key features that need to be supported by the system is given in fig 3.1

### USECASE DIAGRAM

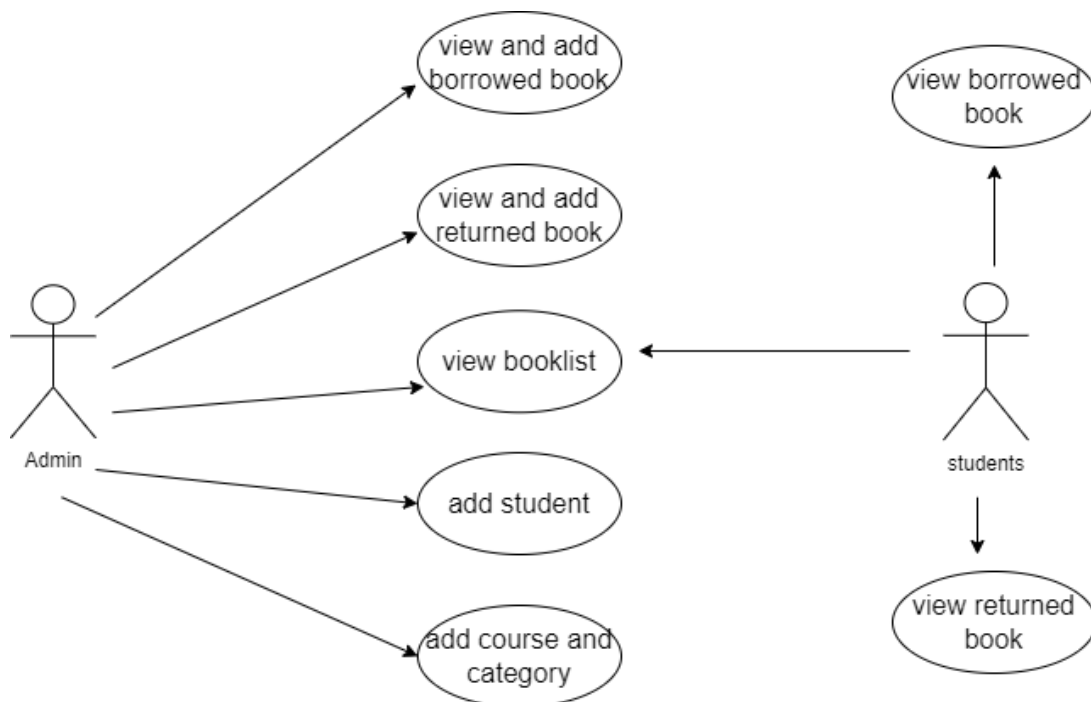


Fig 3.1: Use Case diagram for Library Management System



A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other type of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often showed as stick figures.

## NON-FUNCTIONAL REQUIREMENTS

### Safety Requirements

The database may get crashed at any certain time due to virus or operatingsystem failure. Therefore it is required to take the database backup

### Security Requirements

We are going to develop a secured database for the university .There are different categories of users namely teaching staff, administrator, library staff ,students etc., Depending upon the category of user the access rights are decided. It means if the user is an administrator then he can be able to modify the data, delete, append etc., all other users other than library staff only have the rights to retrieve the information about database.

### Software Quality Attributes

The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database.

### Hardware Constraints

The system requires a database in order to store persistent data. The database should have backup capabilities.

### Software Constraints

The development of the system will be constrained by the availability of required software such as database and development tools.

## Usability

Usability is the main non-functional requirement for a library management system. The UI should be simple enough for everyone to understand and get the relevant information without any special training. Different languages can be provided based on the requirements.

## Accuracy

Accuracy is another important non-functional requirement for the library management system. The data stored about the books and the students should be correct, consistent, and reliable.

## SOFTWARE SPECIFICATION

- Project Type: Web-Based Application
- Front-end Tech: HTML, CSS, BOOTSTRAP, JavaScript
- Database Tool: MySQL
- Back-end Tech: PHP
- OS: Windows 8 and above, Linux and Mac compatible.
- Browser: Internet explorer, Chrome, Firefox, or Safari
- Software: XAMPP

## HARDWARE SPECIFICATION

- Processor: x86 compatible processor with 1.7 GHz Clock Speed
- RAM: 512 MB or greater
- Hard Disk: 20 GB or greater
- Monitor: VGA/SVGA
- Keyboard: 104 keys standard
- Mouse: 2/3 button. Optical/Mechanical.

## USER CHARACTERISTICS

Every user:

- Should be comfortable with basic working of the computer
- Must have basic knowledge of English

## CHAPTER 4

# SYSTEM DESIGN

---

### INTRODUCTION TO SYSTEM DESIGN

System is a collection of an interrelated components that works together to achieve a purpose. System analysis is referred to the systematic examination or detailed study of a system in order to identify problems of the system, and using the information gathered in the analysis stage to recommend improvements or solutions to the system.

System design is an abstract representation of a system component and their relationship and which describe the aggregated functionality and performance of the system. System design is also the plan or blueprint for how to obtain an answer to the question being asked. The design specifies which of the various types of approach. Systems design is the process or art of defining the architecture, components modules, interfaces, and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development.

**Database Design** is a collection of processes that facilitate the designing, development, implementation and maintenance of enterprise data management systems. Properly designed databases are easy to maintain, improve data consistency and are cost effective in terms of disk storage space. The database designer decides how the data elements correlate and what data must be stored. The main objectives of database designing are to produce logical and physical designs models of the proposed database system.

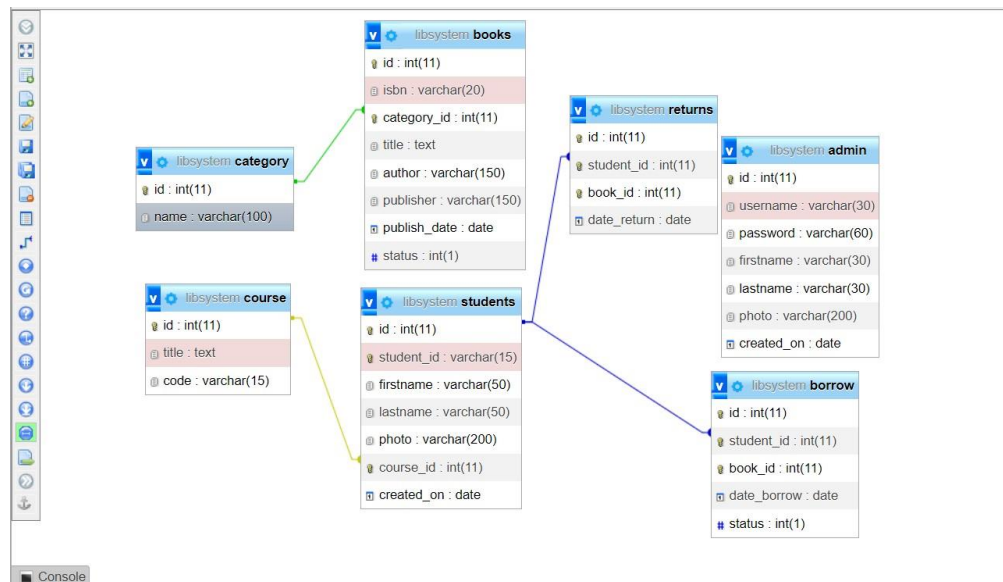
### ATTRIBUTES

Attributes define the properties of a data object and take on one of three different characteristics.

They can be used to:

- Name an instance of a data object.
- Describe the instance.

The following figure 4.1 shows the Relational model for Library Management System.



**Fig 4.1: Relational Model for Library Management System**

The Relational Model was proposed by E.F. Codd to model data in the form of relations or tables. After designing the conceptual model of Database using an ER diagram, we need to convert the conceptual model into the relational model which can be implemented using any RDBMS languages like Oracle SQL, MySQL etc. So, we will see what the Relational Model is. Relational Model represents how data is stored in Relational Databases. A relational database stores data in the form of relations (tables).

## SCHEMA DIAGRAM

A **database schema** is the skeleton structure that represents the logical view of the entire database. A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams.

A **schema diagram** contains entities and the attributes that will define that **schema**. It only shows us the database design. It does not show the actual data of the database. **Schema** can be a single table or it can have more than one table which is related.

The following figure 4.2 shows the Schema for Library Management System.

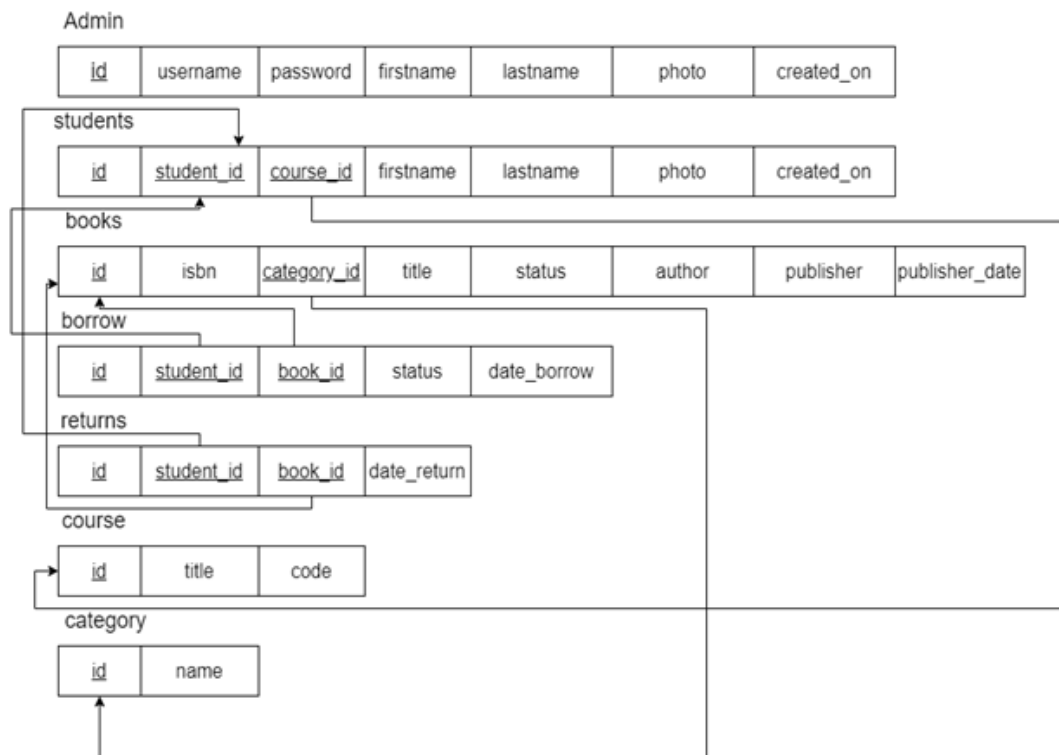


Fig 4.2: Schema diagram for Library Management System

The term "schema" refers to the organization of data as a blueprint of how the database is constructed. These integrity constraints ensure compatibility between parts of the schema. All constraints are expressible in the same language. A database can be considered a structure in realization of the database language. The states of a created conceptual schema are transformed into an explicit mapping, the database schema. This describes how real-world entities are modelled

in the database. All the various tables used are described in the following schema. The necessary Primary keys and the corresponding Foreign keys are also represented.

## ER DIAGRAM

An **Entity–relationship model (ER model)** describes the structure of a database with the help of a diagram, which is known as **Entity Relationship Diagram (ER Diagram)**. An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of ER model are: entity set and relationship set. The ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in a database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.

Here are the geometric shapes and their meaning in an E-R Diagram.

**Rectangle:** Represents Entity sets

**Ellipses:** Attributes

**Diamonds:** Relationship Set

**Lines:** They link attributes to Entity Sets and Entity sets to Relationship Set

**Double Ellipses:** Multivalued Attributes

**Dashed Ellipses:** Derived Attributes

**Double Rectangles:** Weak Entity Sets

**Double Lines:** Total participation of an entity in a relationship set

The following are the Entities and their attributes used in the Library Management System

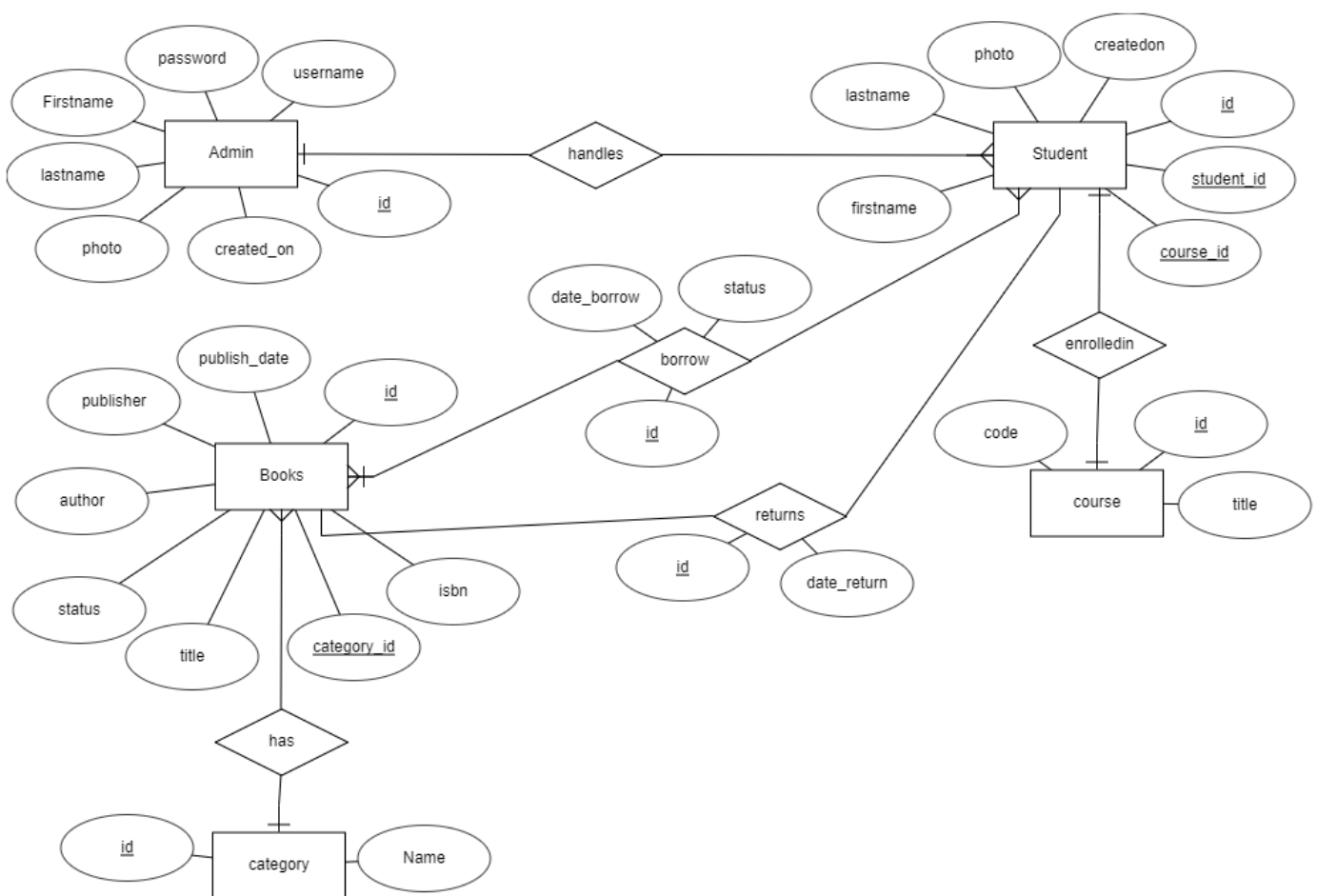
- Admin (username, password, firstname, lastname, photo, created\_on, id)
- Student (firstname, lastname, photo, created\_on, id, student\_id, course\_id)
- Borrow (id, student\_id, book\_id, status, date\_borrow)
- Returns (id, student\_id, book\_id, date\_return)
- Books (id, isbn, category\_id, title, status, author, publisher, publish\_date)
- Course (id, title, code)
- Category (id, name)

The underlined attributes in the above schema represents primary keys.

## RELATIONSHIP BETWEEN ENTITIES

- An Admin can handle n students. Hence the relationship is 1:N.
- M students can borrow N books. Hence the relationship is M:N.
- N number of students can be enrolled per course. Hence the relationship is N:1.
- A category can have N books. Hence the relationship is 1:N.

The following figure 4.3 shows the Entity Relationship for Library Management System.



**Fig 4.3: Entity Relationship Model for Library Management System**

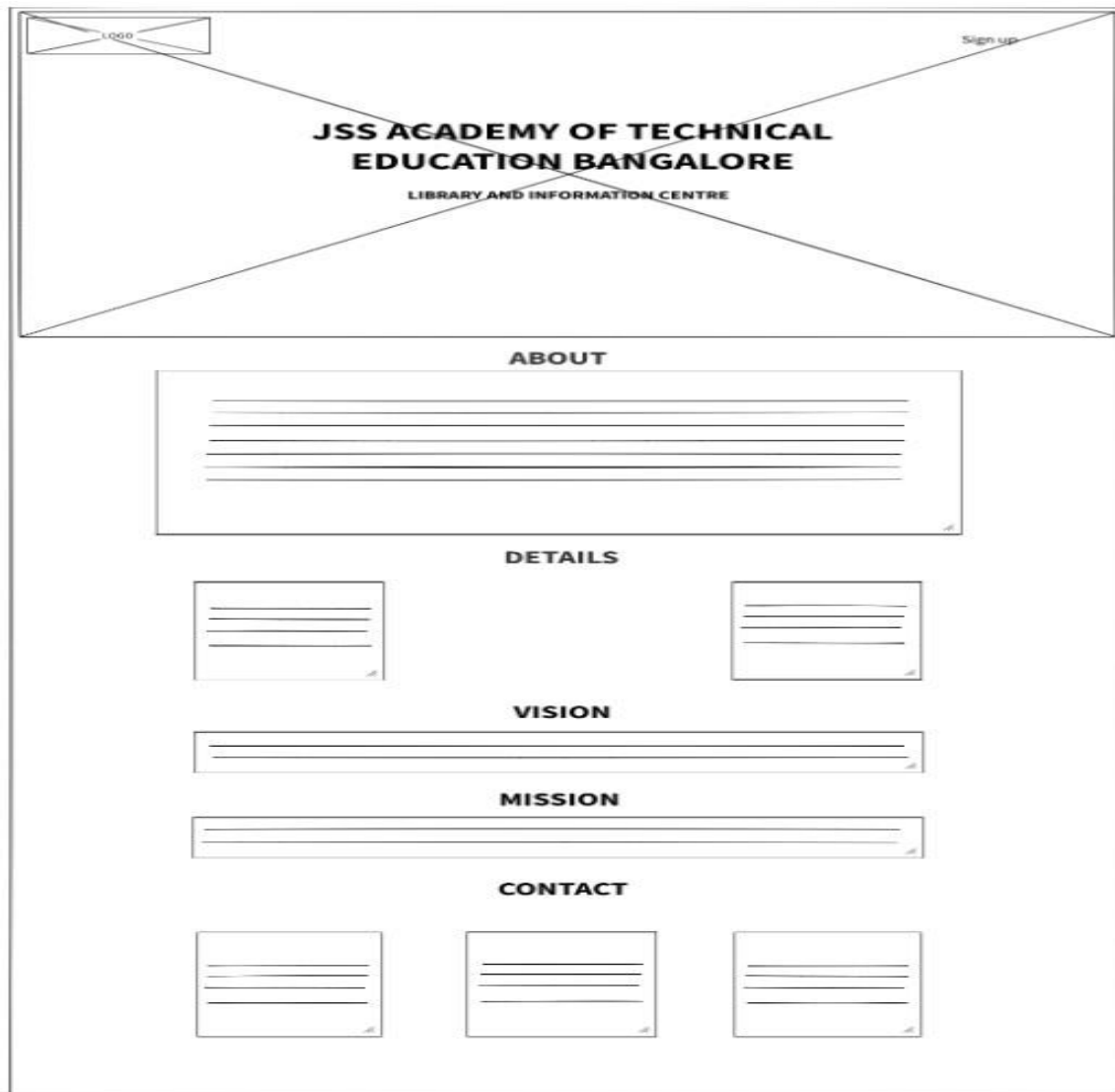
An entity–relationship model or the ER Diagram describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types and specifies relationships that can exist between instances of those entity types.

## CHAPTER 5

# WIREFRAMES

### INTRODUCTION TO WIREFRAMES

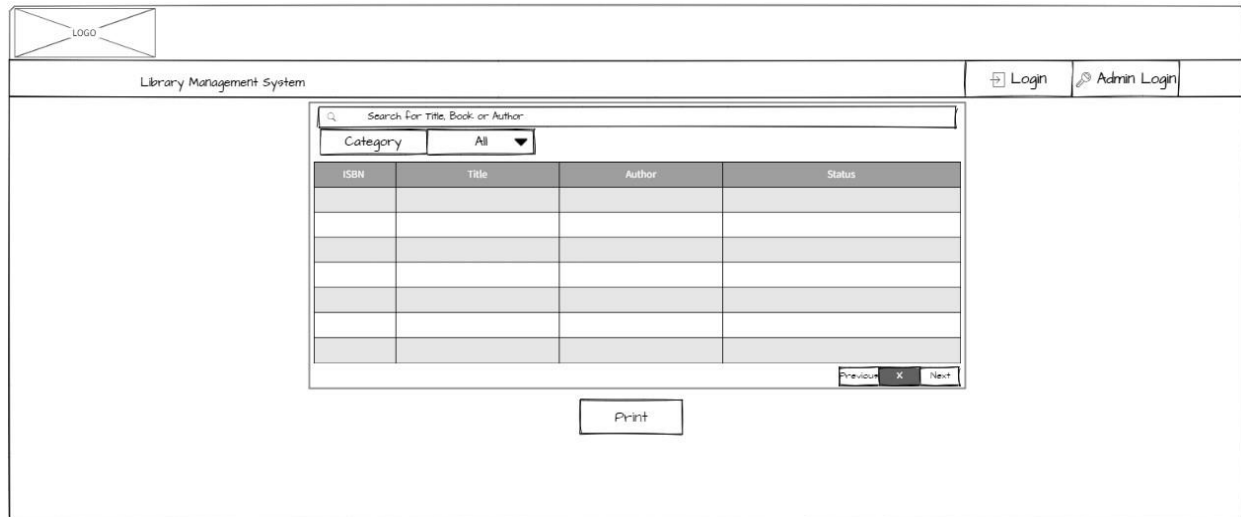
A wireframe is a schematic, a blueprint, useful to help you and your programmers and designers think and communicate about the structure of the software or website you're building. Doing this work, before any code is written and before the visual design is finalized, will save lots of time and painful adjustment work later.



**Fig 5.1: Landing page of Library Management System**

The above Figure 5.1 shows the entry point for the Library Management System website which contains information about the college library and its vision, mission and the contact details.

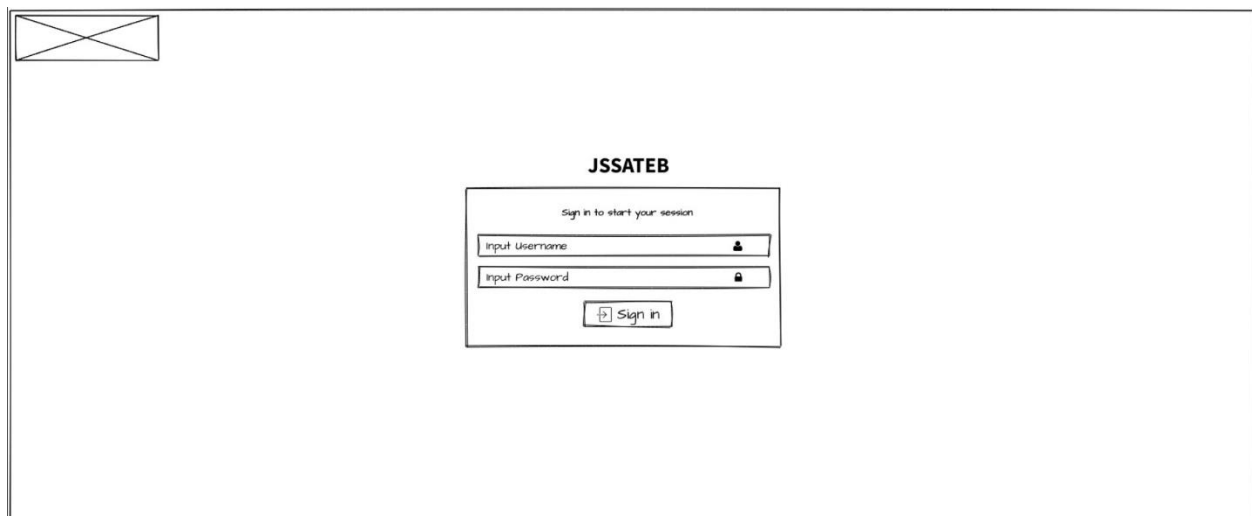




The screenshot shows the home page of a Library Management System. At the top left is a placeholder for a logo. The header contains the text "Library Management System" and two buttons: "Login" and "Admin Login". Below the header is a search bar with the placeholder text "Search For Title, Book or Author". To the left of the search bar is a "Category" dropdown menu currently set to "All". Below the search bar is a table with four columns: "ISBN", "Title", "Author", and "Status". The table has several empty rows. At the bottom right of the table are "Previous", "X", and "Next" buttons. Below the table is a "Print" button.

**Fig 5.2: Home page of Library Management System**

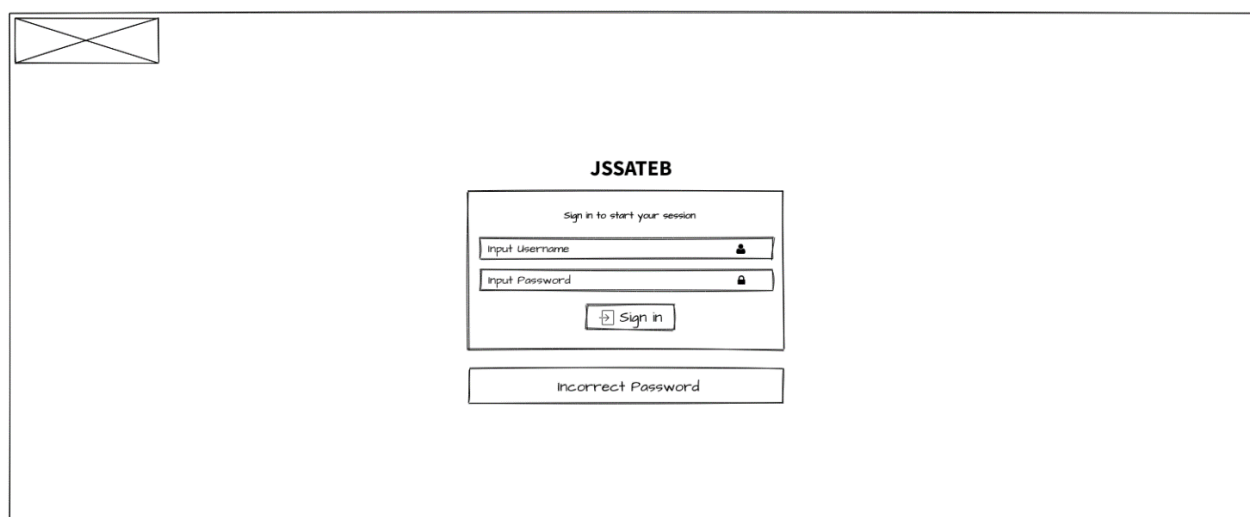
The above figure 5.2 shows the home page which is the default or the front page of this site. It contains the information about the available books and its status. The student or admin can login through this page.



The screenshot shows the admin login page. At the top left is a placeholder for a logo. The page title is "JSSATEB". Below the title is a box containing the text "Sign in to start your session". Inside this box are two input fields: "Input Username" with a user icon and "Input Password" with a lock icon. Below these fields is a "Sign in" button.

**Fig 5.3: Admin login page**

The above figure 5.3 is the login page for admin. This page allows the admin to gain access to Library Management System by entering username and password.



**JSSATEB**

Sign in to start your session

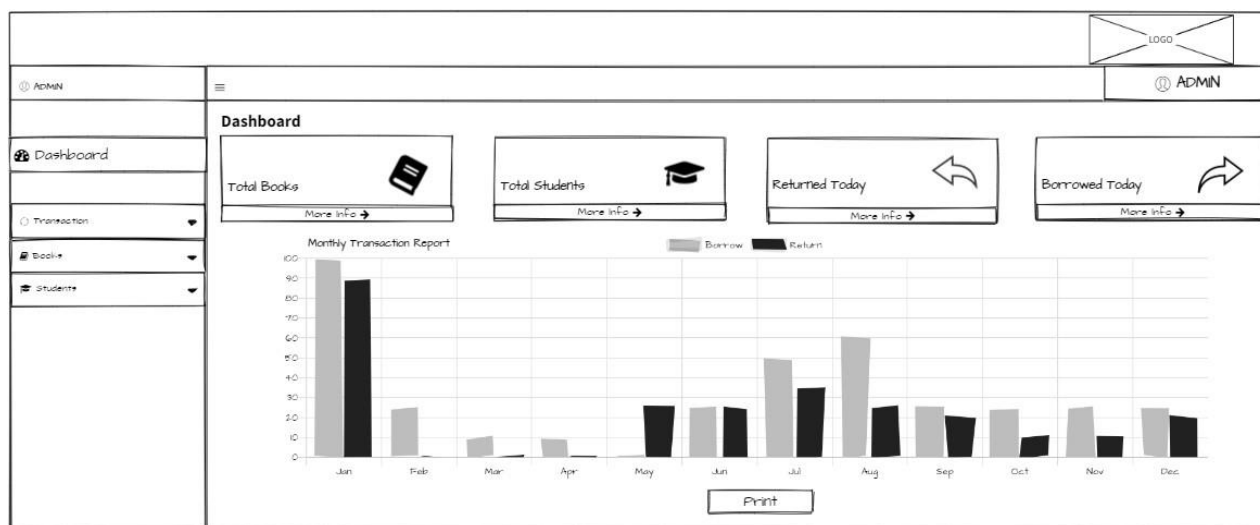
Input Username

Input Password

Incorrect Password

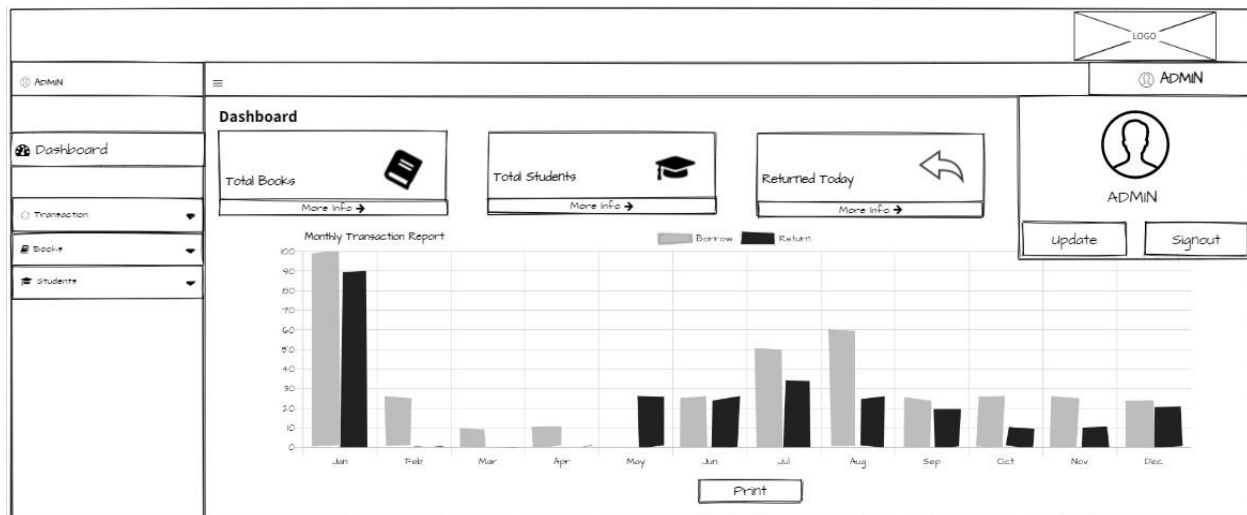
**Fig 5.4: Error notification for incorrect password entered.**

The error notification is generated when the admin enters the incorrect username and password as shown in the figure 5.4.



**Fig 5.5: Admin Dashboard**

The contents of Library Management System are displayed in the Admin Dashboard as shown in the figure 5.5.



**Fig 5.6: Panel to update admin profile or sign out**

This panel allows admin to update his profile or to sign out from this site

The 'Admin Profile' modal form contains the following fields and controls:

- Username:** Text input field.
- Password:** Password input field (masked with dots).
- Firstname:** Text input field.
- Lastname:** Text input field.
- Photo:** File upload area with a 'Choose File' button and 'No File Chosen' text.
- Current Password:** Password input field (masked with dots).
- Buttons:** 'Close' (with an 'X' icon) and 'Save' (with a checkmark icon).

**Fig 5.7: Admin profile updation**

The admin can update the username, profile photo, password and various other details as shown in the figure 5.7.

ADMIN

Dashboard

Transaction

Borrow

Return

Book

Student

Borrow Books

+ Borrow

Show 5 Entries

Date	Student ID	Name	ISBN	Title	Status

Showing 5 of X entries

Previous X Next

Print

**Fig 5.8: Borrow books page**

On clicking borrow option in transaction, we land on borrow books page where admin can view borrowed books with its status and can issue books to the students by entering student ID and ISBN as shown in the figure 5.8.

ADMIN

Dashboard

Transaction

Borrow

Return

Book

Student

Return Books

+ Return

Show 5 Entries

Date	Student ID	Name	ISBN	Title	Status

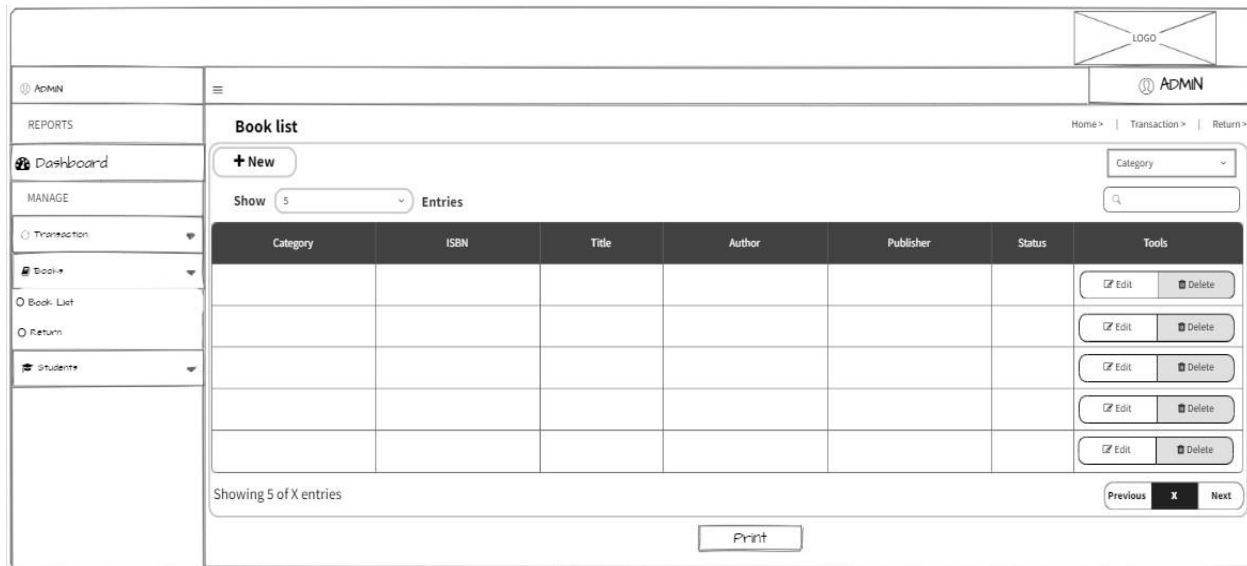
Showing 5 of X entries

Previous X Next

Print

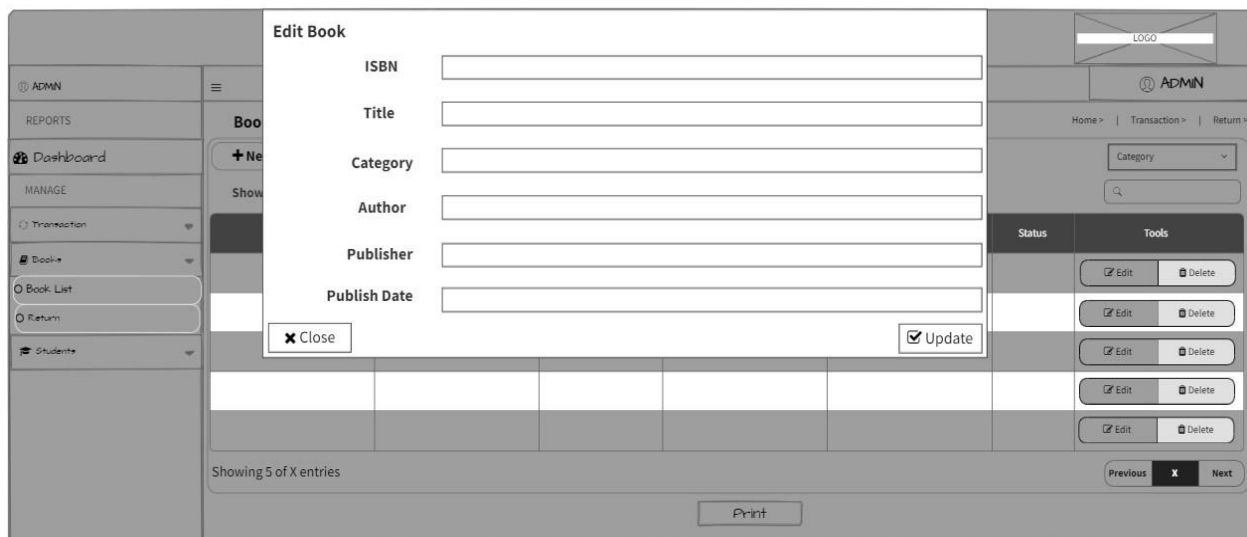
**Fig 5.9: Return books page**

On clicking return option in transaction, we land on return books page where admin can view returned books with its status and can accept the returned books by the students, by entering student ID and ISBN as shown in the figure 5.9.



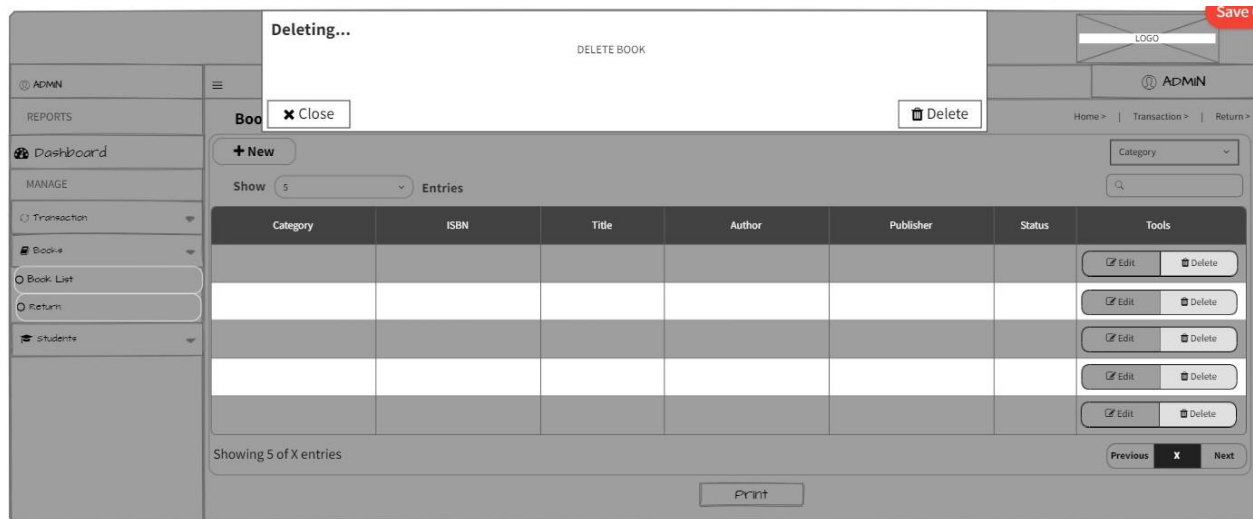
**Fig 5.10: Book list page**

On clicking Book list option in Books, the admin can view Book List and edit the book details or delete books and add new book as shown in the above figure 5.10.



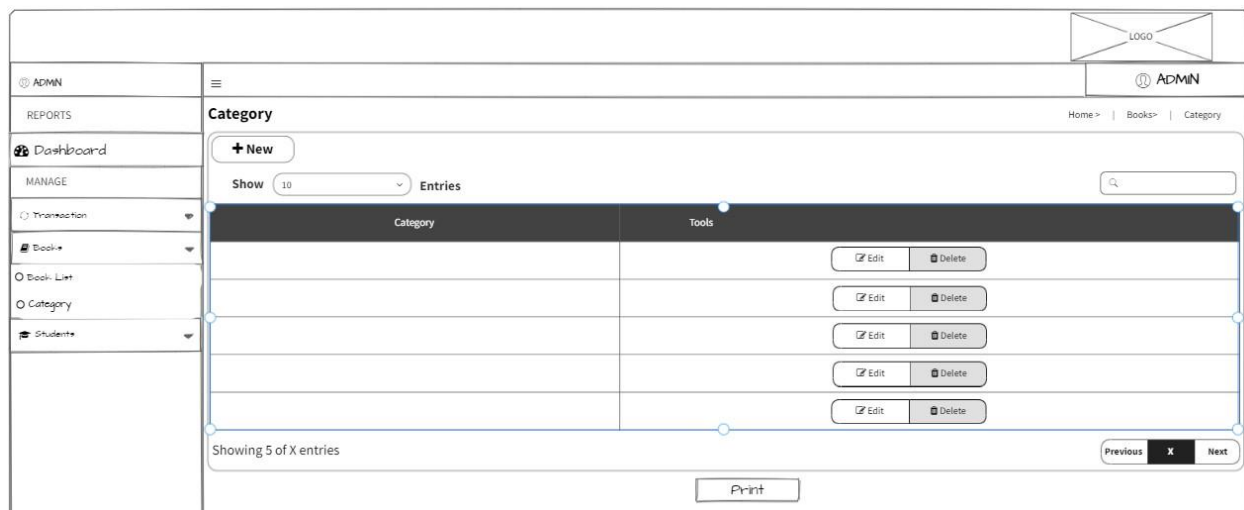
**Fig 5.11: Edit book page**

The various details like ISBN, Title, Category, Author, Publisher and Publisher date of the book can be updated as shown in the figure 5.11.



**Fig 5.12: Delete book page**

The selected book can be deleted as shown in the figure 5.12.



**Fig 5.13: Book Category page**

On clicking category option in Books the admin can view the list of book categories and add new book category as shown in the figure 5.13.

**Edit Category**

Name

**Category**

Show  Entries

Category	Tools
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>

Showing 5 of X entries

**Fig 5.14: Edit category page**

The category of the selected book can be updated as shown in the above figure 5.14.

**Deleting...**

DELETE CATEGORY

**Category**

Show  Entries

Category	Tools
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
	<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>

Showing 5 of X entries

**Fig 5.15: Delete category page**

The selected book's category can be deleted as shown in the figure 5.15.

**Edit Course**

Code

Title

Code	Course Title	Student ID	Tools
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>

Showing 5 of X entries

**Fig 5.16: Edit course page**

On clicking Student list option in Students, the admin can view Student List and edit the Student details or delete student and add new student as shown in the above figure 5.16.

**Deleting...**

DELETE COURSE

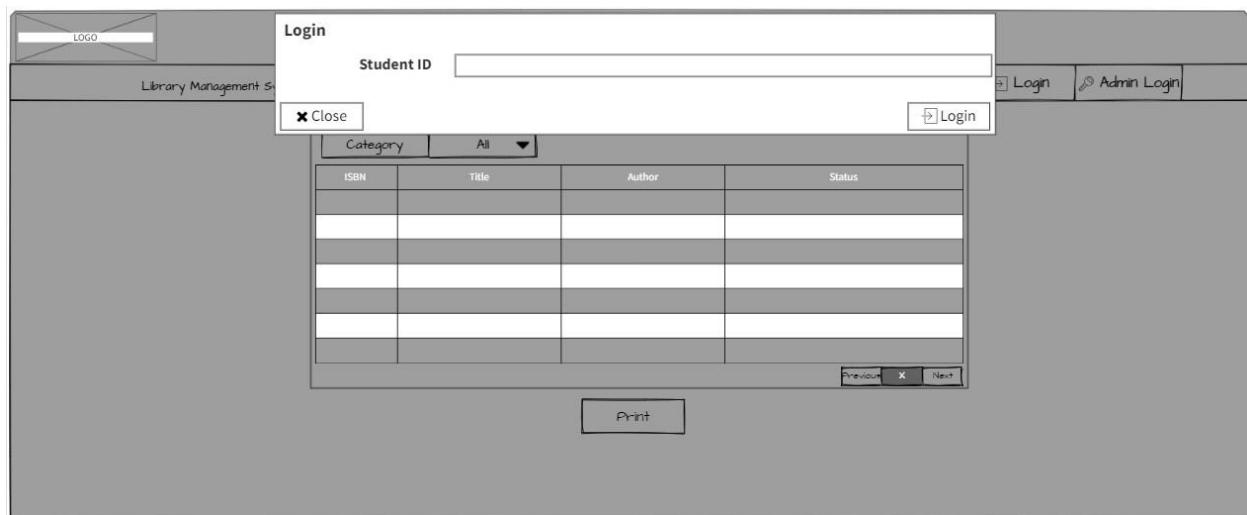
Code	Course Title	Student ID	Tools
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>
			<input checked="" type="button" value="Edit"/> <input type="button" value="Delete"/>

Showing 5 of X entries

**Fig 5.17: Delete course page**

The various details like Student's First name, Last name, Student ID and Course of the book can be updated as shown in the figure 5.17.

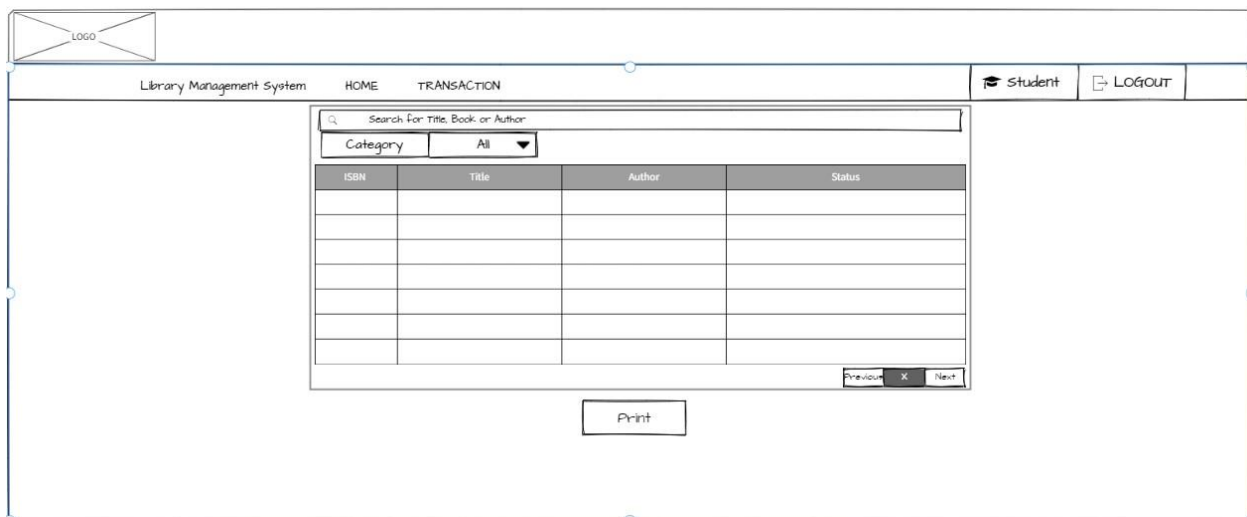




The interface shows a 'Login' modal window. It has a 'Student ID' input field and a 'Login' button. A 'Close' button is also present. In the background, there is a table with columns: ISBN, Title, Author, and Status. The table has 6 empty rows. Below the table are 'Previous', 'Next', and a 'Print' button. The top navigation bar includes a 'LOGO' placeholder, 'Library Management S...', and buttons for 'Login' and 'Admin Login'.

**Fig 5.18: Student Login page**

The student can login through this page as shown in figure 5.18.



The interface shows the 'Book List' page. It features a search bar with the placeholder 'Search For Title, Book, or Author'. Below the search bar is a 'Category' dropdown menu set to 'All'. A table with columns: ISBN, Title, Author, and Status is displayed, containing 6 empty rows. At the bottom of the table are 'Previous', 'Next', and a 'Print' button. The top navigation bar includes a 'LOGO' placeholder, 'Library Management System', 'HOME', 'TRANSACTION', and user controls for 'Student' and 'Logout'.

**Fig 5.19: Book List page**

Student can view book list from this page 5.19.

Library Management System   HOME   TRANSACTION   Student   Logout

**TRANSACTIONS**   Borrow

Show 3 entries

Date	ISBN	Title	Author

Showing 3 of X entries   Previous X Next

Print

**Fig 5.20: Book Transaction page**

Transaction details can be viewed in the above figure 5.20.

## CHAPTER 6

# PROJECT IMPLEMENTATION

---

### INTRODUCTION

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 effectively. The system can be implemented only after thorough testing is done and if it is found to work according to the specification. It involves careful planning, investigating of the current system and its constraints on implementation, design of methods to achieve the change over and an evaluation of change over methods apart from planning. The two major tasks of preparing implementation are education and training of the users and testing of the system.

The following codes will ensure the complete implementation of our design and the project.

### CREATING TABLES

#### TABLE ADMIN

```
CREATE TABLE `admin` (  
  `id` int(11) NOT NULL,  
  `username` varchar(30) NOT NULL,  
  `password` varchar(60) NOT NULL,  
  `firstname` varchar(30) NOT NULL,  
  `lastname` varchar(30) NOT NULL,  
  `photo` varchar(200) NOT NULL,  
  `created_on` date NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
```

DESC admin;

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<b>id</b>	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	<b>username</b>	varchar(30)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3	<b>password</b>	varchar(60)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	4	<b>firstname</b>	varchar(30)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	5	<b>lastname</b>	varchar(30)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	6	<b>photo</b>	varchar(200)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	7	<b>created_on</b>	date			No	None			Change  Drop  More

**Fig 6.1: Table description for Admin**

SELECT\* FROM admin;

		id	username	password	firstname	lastname	photo	created_on
<input type="checkbox"/>	Edit  Copy  Delete	1	ADMIN	\$2y\$10\$zlBQxix2mlIzOahSeEAKFeJ8J9lBnmrkQV4eSJB8dz5...	JSS LIB	ADMIN	jsss.png	2018-05-03

**Fig 6.2: Contents in Admin Table**

## TABLE BOOKS

```
CREATE TABLE `books` (
  `id` int(11) NOT NULL,
  `isbn` varchar(20) NOT NULL,
  `category_id` int(11) NOT NULL,
  `title` text NOT NULL,
  `author` varchar(150) NOT NULL,
  `publisher` varchar(150) NOT NULL,
  `publish_date` date NOT NULL,
  `status` int(1) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
```

DESC books;

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<b>id</b>	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	<b>isbn</b>	varchar(20)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3	<b>category_id</b>	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	4	<b>title</b>	text	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	5	<b>author</b>	varchar(150)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	6	<b>publisher</b>	varchar(150)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	7	<b>publish_date</b>	date			No	None			Change  Drop  More
<input type="checkbox"/>	8	<b>status</b>	int(1)			No	None			Change  Drop  More

Fig 6.3: Table description for Books

SELECT\* FROM books;

			id	isbn	category_id	title	author	publisher	publish_date	status
<input type="checkbox"/>	Edit  Copy  Delete		4	0021	5	Effective C++	Scott Meyers	Hoobstank Publishers	2020-06-03	1
<input type="checkbox"/>	Edit  Copy  Delete		5	01	1	Software Engineering	Ian Sommerville	Jacobs Publisher	2018-05-07	1
<input type="checkbox"/>	Edit  Copy  Delete		6	002	5	Python Cookbook	Brain K.JOnes	Jacobs Publisher	2020-06-01	0
<input type="checkbox"/>	Edit  Copy  Delete		7	005	1	Machinery Handbook	Erik Oberg	Jacobs Publisher	2019-04-03	1
<input type="checkbox"/>	Edit  Copy  Delete		8	555	3	A Brief History of Time	Stephen Hawkings	Jacobs Publisher	2011-06-09	1
<input type="checkbox"/>	Edit  Copy  Delete		9	123	5	Java 2	Herbert	Demo Publisher	2009-05-15	0

Fig 6.4: Contents in books table

## TABLE BORROW

```
CREATE TABLE `borrow` (
  `id` int(11) NOT NULL,
  `student_id` int(11) NOT NULL,
  `book_id` int(11) NOT NULL,
  `date_borrow` date NOT NULL,
  `status` int(1) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
```

DESC borrow;

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	student_id	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	3	book_id	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	4	date_borrow	date			No	None			Change  Drop  More
<input type="checkbox"/>	5	status	int(1)			No	None			Change  Drop  More

Fig 6.5: Table description for Borrow

SELECT\* FROM Borrow

					id	student_id	book_id	date_borrow	status
<input type="checkbox"/>	Edit	Copy	Delete		17	3	1	2021-05-04	0
<input type="checkbox"/>	Edit	Copy	Delete		18	3	2	2021-05-04	1
<input type="checkbox"/>	Edit	Copy	Delete		19	5	3	2021-06-26	0
<input type="checkbox"/>	Edit	Copy	Delete		23	6	7	2021-06-26	0
<input type="checkbox"/>	Edit	Copy	Delete		24	6	4	2021-06-26	0

Fig 6.6: Contents in return table

## TABLE CATEGORY

```
CREATE TABLE `category` (
  `id` int(11) NOT NULL,
  `name` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
```

DESC category;

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<b>id</b>	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	<b>name</b>	varchar(100)	latin1_swedish_ci		No	None			Change  Drop  More

Fig 6.7: Table description for Category

SELECT\* FROM category;

					id	name
<input type="checkbox"/>	Edit	Copy	Delete		1	Engineering
<input type="checkbox"/>	Edit	Copy	Delete		2	Mathematics
<input type="checkbox"/>	Edit	Copy	Delete		3	Science and Technology
<input type="checkbox"/>	Edit	Copy	Delete		4	History
<input type="checkbox"/>	Edit	Copy	Delete		5	IT Programming

Fig 6.8: Contents in category table

## TABLE COURSE

```
CREATE TABLE `course` (
  `id` int(11) NOT NULL,
  `title` text NOT NULL,
  `code` varchar(15) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
```

DESC course;

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<b>id</b>	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	<b>title</b>	text	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3	<b>code</b>	varchar(15)	latin1_swedish_ci		No	None			Change  Drop  More

Fig 6.9: Table description for Course

SELECT\* FROM table;



				id	title	code
<input type="checkbox"/>				1	Information Science Engineering	ISE
<input type="checkbox"/>				2	Computer Science Engineering	CSE
<input type="checkbox"/>				3	Electronics And Communication Engineering	ECE

Fig 6.10: Contents in course table

## TABLE RETURNS

```
CREATE TABLE `returns` (
  `id` int(11) NOT NULL,
  `student_id` int(11) NOT NULL,
  `book_id` int(11) NOT NULL,
  `date_return` date NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
```

DESC returns;
















#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 id 	int(11)			No	None		AUTO_INCREMENT	 Change  Drop  More
<input type="checkbox"/>	2 student_id 	int(11)			No	None			 Change  Drop  More
<input type="checkbox"/>	3 book_id 	int(11)			No	None			 Change  Drop  More
<input type="checkbox"/>	4 date_return	date			No	None			 Change  Drop  More

Fig 6.11: Table description for Returns

SELECT\* FROM returns;

				id	student_id	book_id	date_return
<input type="checkbox"/>				3	3	2	2021-05-04
<input type="checkbox"/>				4	3	2	2021-05-04
<input type="checkbox"/>				5	6	4	2021-06-26

Fig 6.12: Contents in Returns table



## TABLE STUDENTS

```
CREATE TABLE `students` (
  `id` int(11) NOT NULL,
  `student_id` varchar(15) NOT NULL,
  `firstname` varchar(50) NOT NULL,
  `lastname` varchar(50) NOT NULL,
  `photo` varchar(200) NOT NULL,
  `course_id` int(11) NOT NULL,
  `created_on` date NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
```

DESC students;

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 <b>id</b> 🔑	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2 <b>student_id</b> 🔑	varchar(15)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3 <b>firstname</b>	varchar(50)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	4 <b>lastname</b>	varchar(50)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	5 <b>photo</b>	varchar(200)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/>	6 <b>course_id</b> 🔑	int(11)			No	None			Change  Drop  More

Fig 6.13: Table description for Students

SELECT\* FROM students;

				id	student_id	firstname	lastname	photo	course_id	created_on
<input type="checkbox"/>	Edit	Copy	Delete	3	1JS21IS001	Kavita	K		1	2021-05-04
<input type="checkbox"/>	Edit	Copy	Delete	4	1JS21IS002	Karan	C		2	2021-05-04
<input type="checkbox"/>	Edit	Copy	Delete	5	1JS21IS030	Bharath	G		1	2021-06-26
<input type="checkbox"/>	Edit	Copy	Delete	6	1JS21CS015	Suresh	M		1	2021-06-26

Fig 6.14: Contents in students table

## QUERIES

The below mentioned are all the queries used to perform various tasks in MySQL such as insert, delete, update, joins and triggers. A short description of the query is also provided.

### BOOKS TABLE

**Query:**

```
SELECT *, books.id AS bookid FROM books LEFT JOIN category ON  
category.id=books.category_id WHERE books.id = '$id';
```

**Description:**

*SQL JOINS* are used to retrieve data from multiple tables. A SQL JOIN is performed whenever two or more tables are listed in a SQL statement. In this query, we aim to fetch the data using the concepts of JOINS. The query fetches data from two tables BOOKS and CATEGORY based on *category\_id*, where *category\_id* is foreign key in BOOKS. This query uses the concept of *Natural Joins*.

### STUDENTS TABLE

**Query:**

```
SELECT *, students.id AS studid FROM students LEFT JOIN course ON  
course.id=students.course_id WHERE students.id = '$id';
```

**Description**

*SQL JOINS* are used to retrieve data from multiple tables. A SQL JOIN is performed whenever two or more tables are listed in a SQL statement. In this query, we take the JOIN of three tables from STUDENTS, COURSE.

## BORROW

### Query:

```
SELECT * FROM borrow LEFT JOIN books ON books.id=borrow.book_id WHERE student_id = '$stuid' ORDER BY date_borrow DESC";
```

### Description:

The SELECT statement in MySQL is used to **fetch data from one or more tables**. We can retrieve records of all fields or specified fields that match specified criteria using this statement. It can also work with various scripting languages such as PHP, Ruby, and many more. The ORDER BY keyword is used to *sort the result-set* in ascending or descending order. The above query is a simple SELECT statement which fetches all the data stored in BORROW and orders them in descending order.

## RETURN

### Query:

```
"SELECT * FROM returns LEFT JOIN books ON books.id=returns.book_id WHERE student_id = '$stuid' ORDER BY date_return DESC";
```

### Description:

The SELECT statement in MySQL is used to **fetch data from one or more tables**. We can retrieve records of all fields or specified fields that match specified criteria using this statement. It can also work with various scripting languages such as PHP, Ruby, and many more. The ORDER BY keyword is used to *sort the result-set* in ascending or descending order. The above query is a simple SELECT statement which fetches all the data stored in RETURN and orders them in descending order.

## TRIGGERS

A trigger is a stored procedure in the database which automatically invokes whenever a special event in the database occurs. For example, a trigger can be invoked when a row is inserted into a specified table or when certain table columns are being updated.

**Syntax:**

```
create trigger [trigger_name]
[before | after]
{insert | update | delete}
on [table_name]
[for each row]
[trigger_body]
```

**TRIGGER FOR BORROW**

```
CREATE TRIGGER `upstatus` AFTER INSERT ON `borrow` FOR EACH ROW BEGIN
UPDATE books set status= 0 ;
END
```

**Description:**

In this trigger we aim to update the status of the data that will be updated after the book is borrowed from the library. Here, we have used AFTER INSERT ON BORROW which will update the values of the BOOKS table as shown below.

**TRIGGER FOR RETURNS**

```
CREATE TRIGGER `downstatus` AFTER INSERT ON `returns` FOR EACH ROW BEGIN
UPDATE books set status= 1 ;
END
```

**Description:**

In this trigger we aim to update the status of the data that will be updated after the book is returned from the library. Here, we have used AFTER INSERT ON RETURNS which will update the values of the BOOKS table as shown below.

## CHAPTER 7

# SYSTEM TESTING

---

### INTRODUCTION

Testing plays a vital role in the success of the system. System testing makes a logical assumption that if all parts of the system are correct, the goal will be successfully achieved. Once program code has been developed, testing begins. The testing process focuses on the logical internals of the software, ensuring that all statements have been tested, and on the functional externals, that is conducted tests to uncover errors and ensure that defined input will produce actual results that agree with required results. Broadly speaking, there are at least three levels of testing: unit testing, integration testing, and system testing

### TYPES OF TESTING

#### Unit testing

Unit testing refers to tests that verify the functionality of a specific section of code, usually at the function level. In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors. These types of tests are usually written by developers as they work on code (white-box style), to ensure that the specific function is working as expected. One function might have multiple tests, to catch corner cases or other branches in the code. Unit testing alone cannot verify the functionality of a piece of software, but rather is used to ensure that the building blocks of the software work independently from each other.

Unit testing is a software development process that involves a synchronized application of a broad spectrum of defect prevention and detection strategies in order to reduce software development risks, time, and costs. It is performed by the software developer or engineer during the construction phase of the software development lifecycle. Unit testing aims to eliminate construction errors before code is promoted to additional testing; this strategy is intended to increase the quality of the resulting software as well as the efficiency of the overall development process.

SL.NO	Test Cases	Test Results
1	Database	Successful
2	Foreign key constraint	Successful
3	Triggers	Successful
4	Relational Schema	Successful
5	Validation of inputs	Successful
6	Database Updation	Successful
7	Front-end View	Successful

**Table 7.1: Unit test cases for Library Management System**

### Integration Testing

Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an iterative way or all together ("big bang"). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed.

Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.

### Admin Login Page :

SL.NO	Test Case	Expected Result	Test Result
1	On clicking SignIn	The user's username and password will be authenticated and it should redirect to the dashboard.	Successful

**Table 7.2: Integration test cases for admin login**

**Book :**

SL.NO	Test Case	Expected Result	Test Result
1	On clicking new button	At first user have to fill all fields with proper data, if any error occurs print message .	Successful
2	On click of delete button	This deletes the details of book by using ISBN.	Successful
3	On click of edit button	Modified records are updated.	Successful
4	On click of search button	Displays the details of book which matches the search key.	Successful

**Table 7.3: Integration test cases for Book****Return page:**

SL.NO	Test Case	Expected Result	Test Result
1	On clicking return button	At first user have to fill all fields with proper data, if any error occurs print message .	Successful
2	On click of search button	Displays the details of students which matches the search key.	Successful

**Table 7.4: Integration test cases for return page****Borrow page:**

SL.NO	Test Case	Expected Result	Test Result
1	On clicking borrow button	At first user must fill all fields with proper data, if any error occurs print message.	Successful
2	On click of search button	Displays the details of students which matches the search key.	Successful

**Table 7.5: Integration test cases for Borrow page**

## System Testing

System testing tests a completely integrated system to verify that the system meets its requirements. For example, a system test might involve testing a logon interface, then creating and editing an entry, plus sending or printing results, followed by summary processing or deletion (or archiving) of entries, then logoff.

SL.NO	Test Case	Expected Result	Test Result
1	Upon clicking dashboard	Renders dashboard page to the home.	Successful
2	Upon clicking Borrow	Borrow books page opens.	Successful
3	Upon clicking Return	Return books page opens.	Successful
4	Upon clicking Book List	Book page will be opened.	Successful
5	Upon clicking category	Category page will open.	Successful
6	Upon clicking student	Student page will open.	Successful
7	Upon clicking course	Course page opens.	Successful

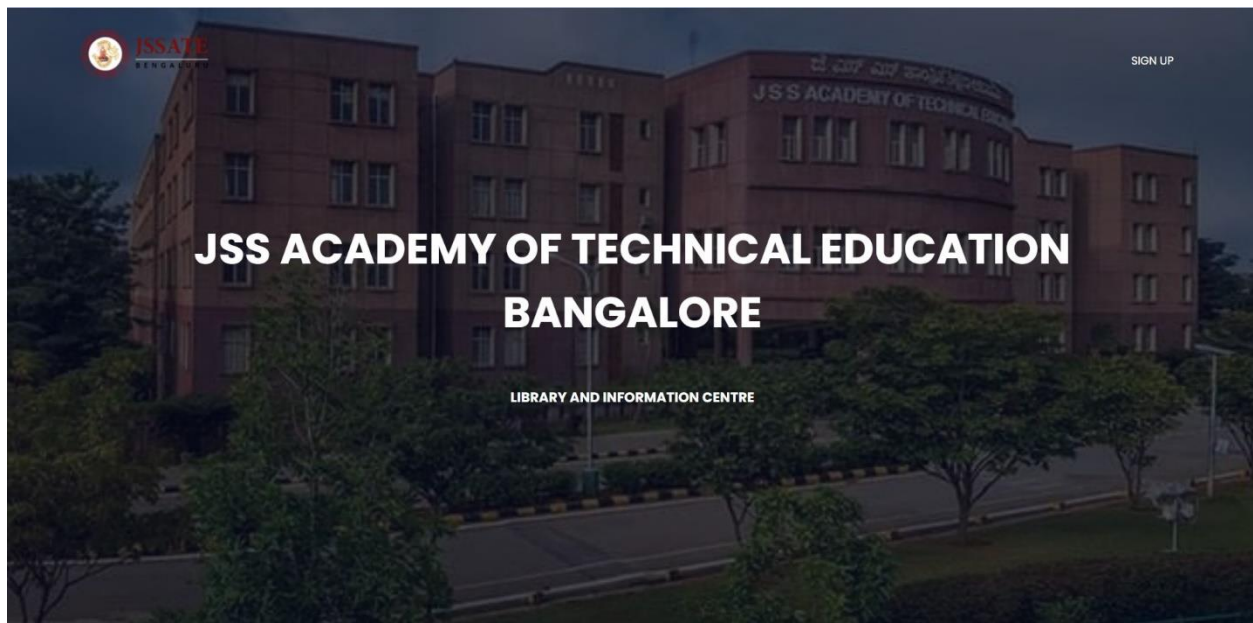
**Table 7.6: System test cases for Library Management System**



# CHAPTER 8

## RESULTS AND DISCUSSIONS

### 1. Landing page of Library Management System:

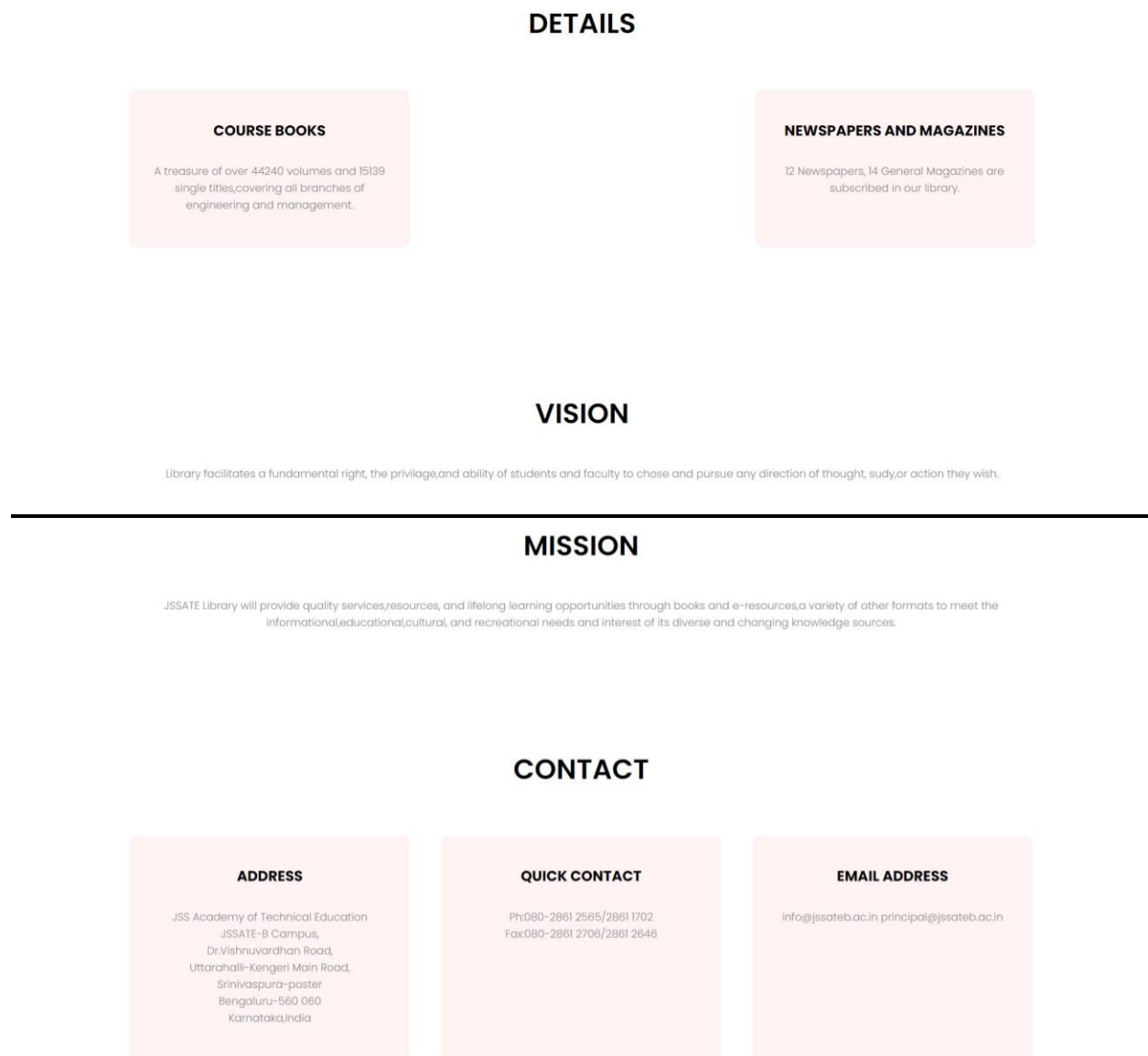


#### ABOUT

The JSSATE-Bangalore Library is a veritable feast of knowledge and is available to students and faculty. It is housed presently in ground floor of the college building (B Block) with 2018.18-sq.mts. carpet area. It has a comprehensive collection of literature predominantly related to engineering, management, and its allied subjects to meet the present and future information needs of the users.

The Library has a collection of about 44,240 documents. The Library subscribes to important technical journals. The Library also has access to good number of E-Resources from renowned publishers VIZ, Elsevier, Springer Nature, Emerald (Management), Taylor and Francis, and Map Systems Digital Library platform etc. E-Resources can be access anywhere in the campus through Wi-Fi and LAN.

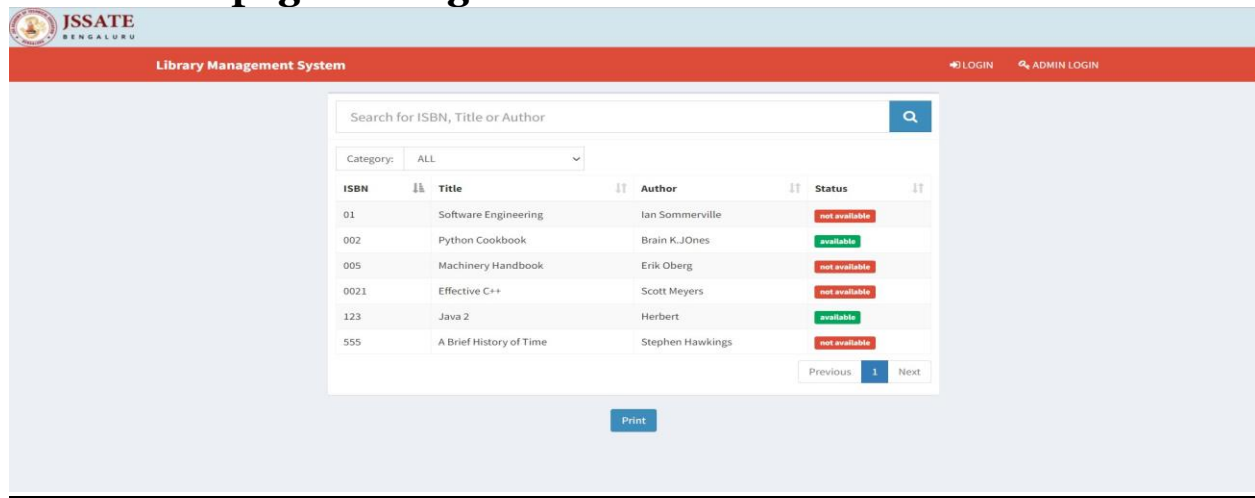
The infrastructure has a centralized Library with open access facility for students. There is also a book bank facility for needy students to borrow the books for the entire semester and we have a Book Bank facility for socially backward Students & they can borrow 5 books for the entire semester. Library has LIBSOFT Library management Software to facilitate smooth functioning of the Library. Digital Library is established to access on line database and online Journals. Air-conditioned Audio/Visual room also setup for seminars/workshops. Discussion rooms (4 no's) with white board facility is also provided. Photocopying/printout facilities are available; Departmental Libraries are also setup by all the major departments



**Fig 8.1: Landing page of Library Management System**

The above Figure 8.1 shows the entry point for the Library Management System website which contains information about the college library and its vision, mission and the contact details.

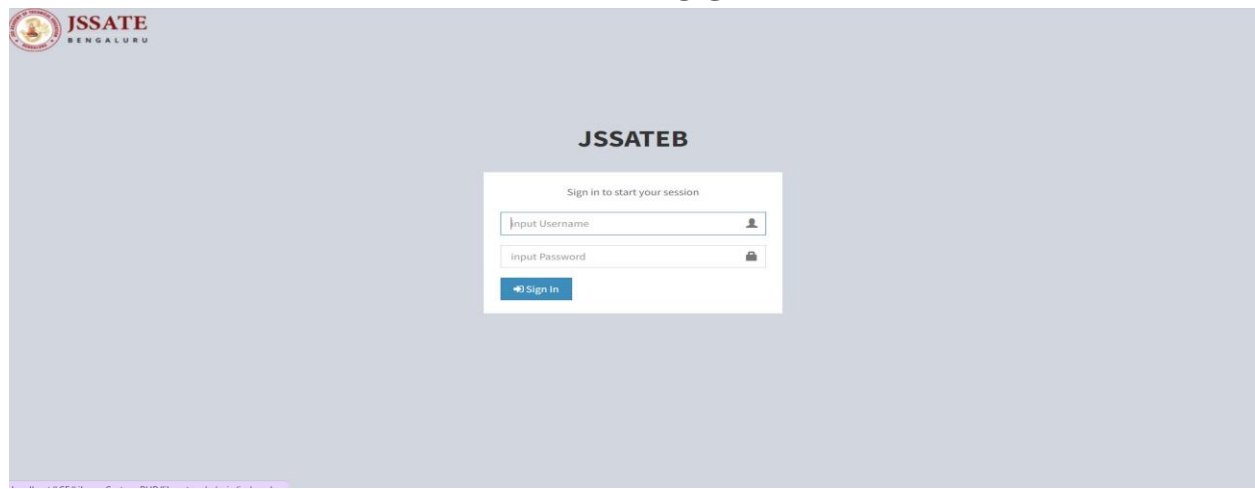
## 2. Home page for Login



**Fig 8.2: Home page of Library Management System**

The above figure 8.2 shows the home page which is the default or the front page of this site. It contains the information about the available books and its status. The student or admin can login through this page.

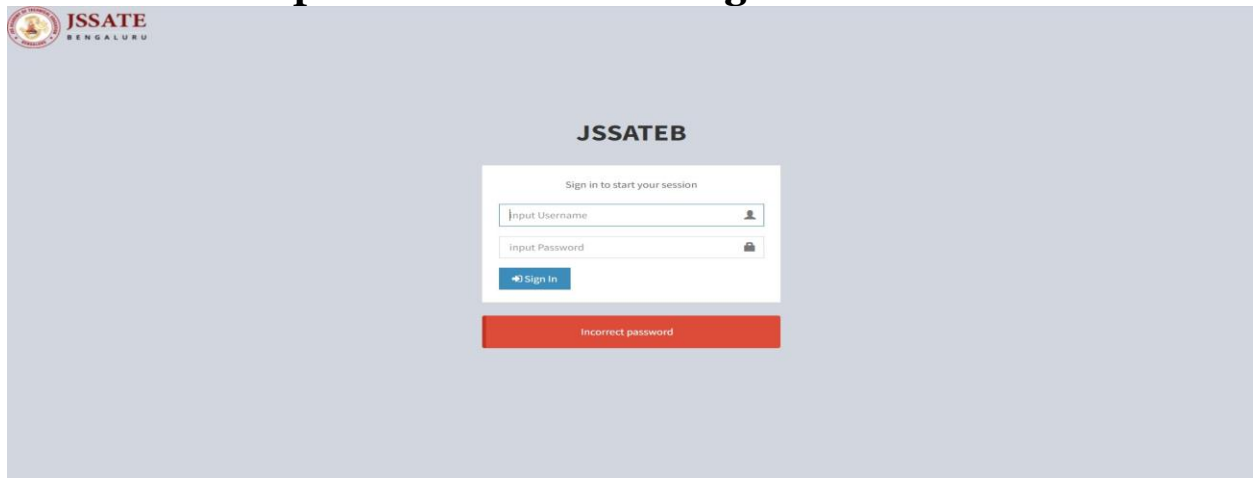
## ADMIN LOGIN



**Fig 8.3: Admin login page**

The above figure 8.3 is the login page for admin. This page allows the admin to gain access to Library Management System by entering username and password.

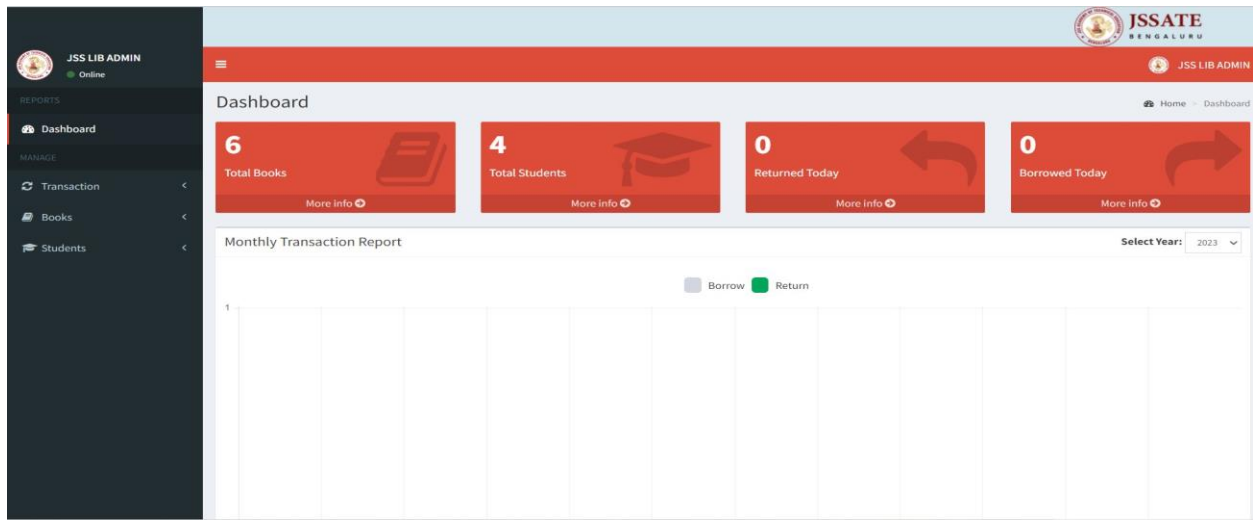
## 1. Incorrect password for admin login



**Fig 8.4: Error notification for incorrect password entered.**

The error notification is generated when the admin enters the incorrect username and password as shown in the figure 8.4.

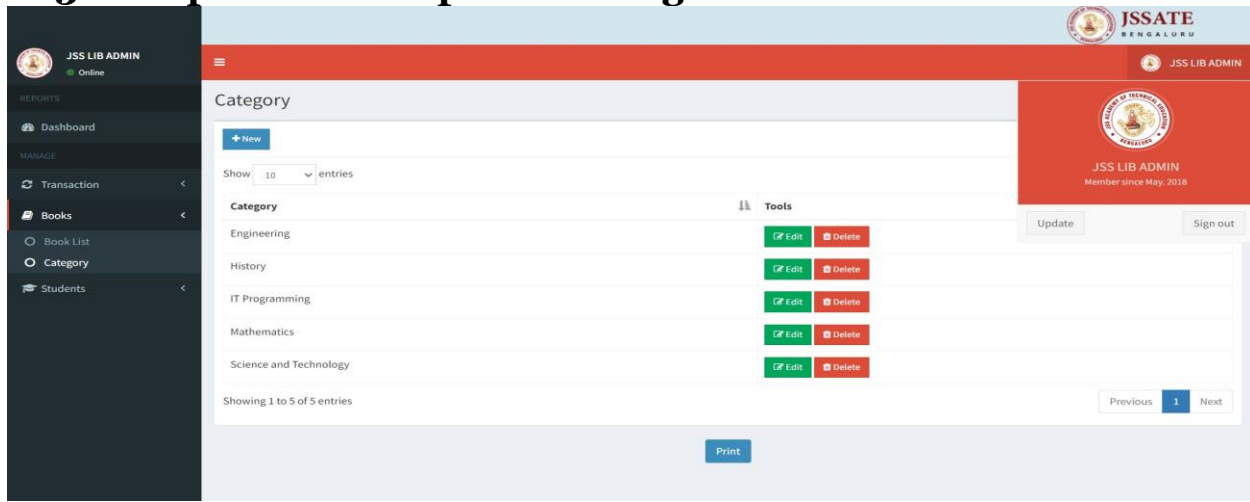
## 2. Admin Dashboard



**Fig 8.5: Admin Dashboard**

The contents of Library Management System are displayed in the Admin Dashboard as shown in the figure 8.5.

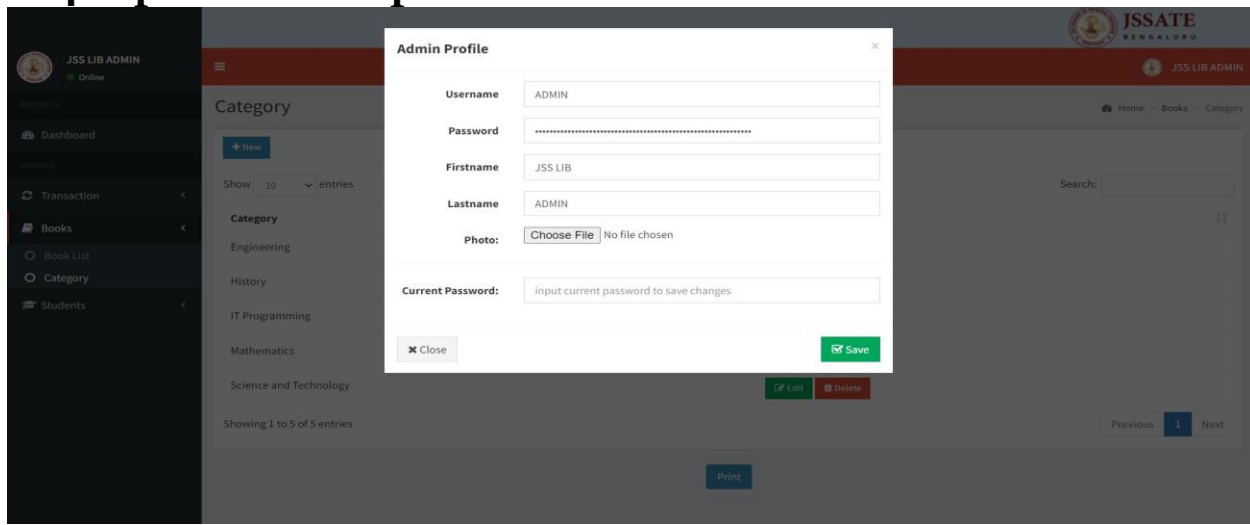
### 3. To update admin profile or sign out



**Fig 8.6: Panel to update admin profile or signout**

This panel allows admin to update his profile or to sign out from this site.

### 4. Update Admin profile



**Fig 8.7: Admin profile updation**

The admin can update the username, profile photo, password and various other details as shown in the figure 8.7.

## 5. On clicking Transaction → Borrow

**Borrow Books**

Home > Transaction > Borrow

[+ Borrow](#)

Show  entries

Search:

Date	Student ID	Name	ISBN	Title	Status
Jun 26, 2021	1JS21IS030	Bharath G			not returned
Jun 26, 2021	1JS21CS015	Suresh M	005	Machinery Handbook	not returned
Jun 26, 2021	1JS21CS015	Suresh M	0021	Effective C++	not returned
May 04, 2021	1JS21IS001	Kavita K			not returned
May 04, 2021	1JS21IS001	Kavita K			returned

Showing 1 to 5 of 5 entries

Previous **1** Next

[Print](#)

**Fig 8.8: Borrow books page**

On clicking borrow option in transaction, we land on borrow books page where admin can view borrowed books with its status and can issue books to the students by entering student ID and ISBN as shown in the figure 8.8.

## 6. On clicking Transaction → Return

**Return Books**

Home → Transaction → Return

+ Returns

Show 10 entries

Search:

Date	Student ID	Name	ISBN	Title
Jun 26, 2021	IJS21CS015	Suresh M	0021	Effective C++
May 04, 2021	IJS21IS001	Kavita K		
May 04, 2021	IJS21IS001	Kavita K		

Showing 1 to 3 of 3 entries

Previous 1 Next

Print

© 2023 - Library Management System

**Fig 8.9: Return books page**

On clicking return option in transaction, we land on return books page where admin can view returned books with its status and can accept the returned books by the students, by entering student ID and ISBN as shown in the figure 8.9.

## 7. On clicking Books → Book List

**Book List**

Home → Books → Book List

+ New

Category: ALL

Show 10 entries

Search:

Category	ISBN	Title	Author	Publisher	Status	Tools
Engineering	01	Software Engineering	Ian Sommerville	Jacobs Publisher	borrowed	[Edit] [Delete]
Engineering	005	Machinery Handbook	Erik Oberg	Jacobs Publisher	borrowed	[Edit] [Delete]
IT Programming	0021	Effective C++	Scott Meyers	Hoobstank Publishers	borrowed	[Edit] [Delete]
IT Programming	002	Python Cookbook	Brain K.Jones	Jacobs Publisher	available	[Edit] [Delete]
IT Programming	123	Java 2	Herbert	Demo Publisher	available	[Edit] [Delete]
Science and Technology	555	A Brief History of Time	Stephen Hawking	Jacobs Publisher	borrowed	[Edit] [Delete]

Showing 1 to 6 of 6 entries

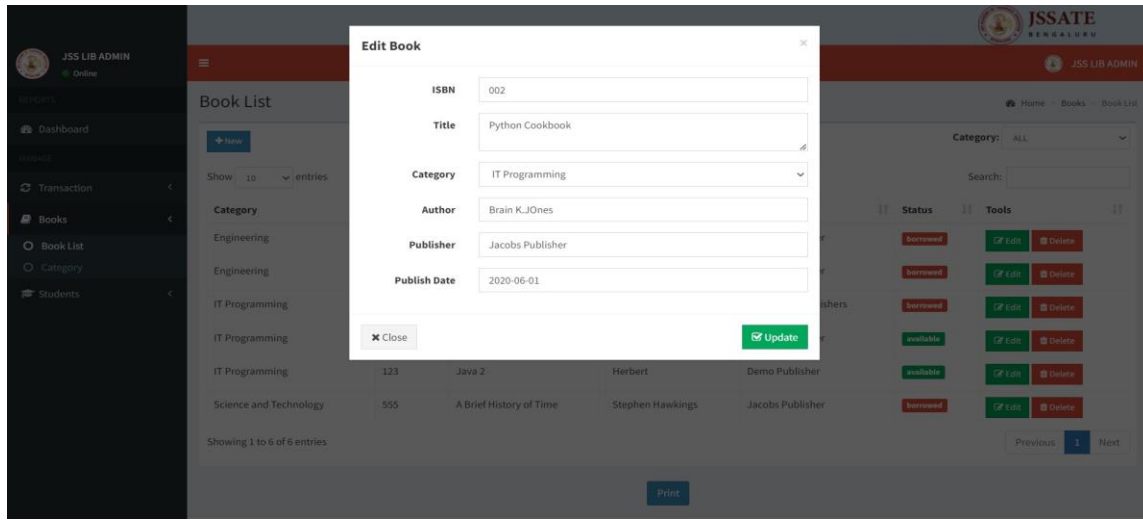
Previous 1 Next

Print

**Fig 8.10: Book list page**

On clicking Book list option in Books, the admin can view Book List and edit the book details or delete books and add new book as shown in the above figure 8.10.

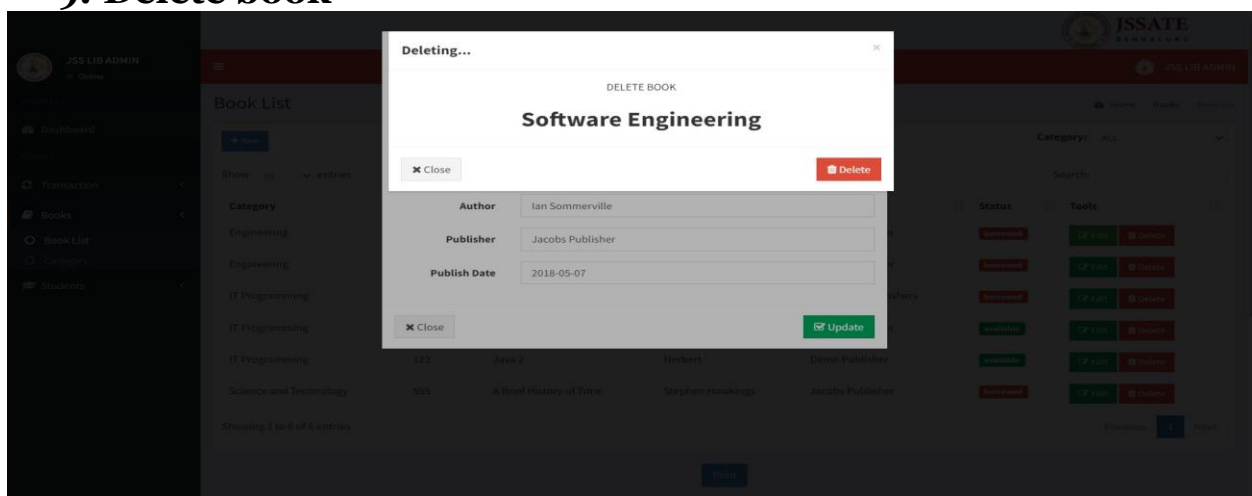
## 8. Edit book



**Fig 8.11: Edit book page**

The various details like ISBN, Title, Category, Author, Publisher and Publisher date of the book can be updated as shown in the figure 8.11.

## 9. Delete book

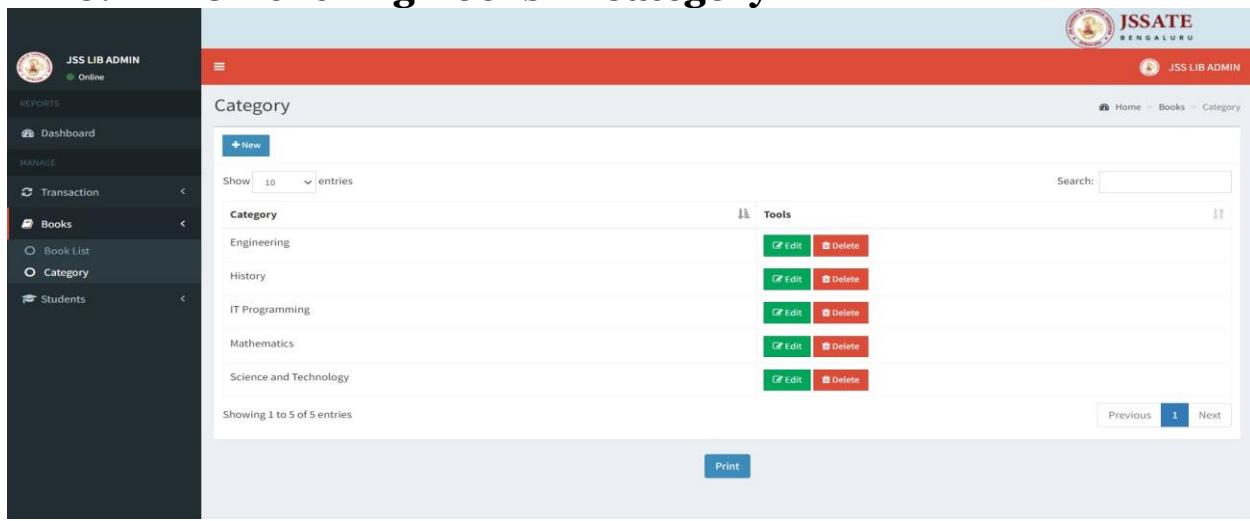


**Fig 8.12: Delete book page**

The selected book can be deleted as shown in the figure 8.12.



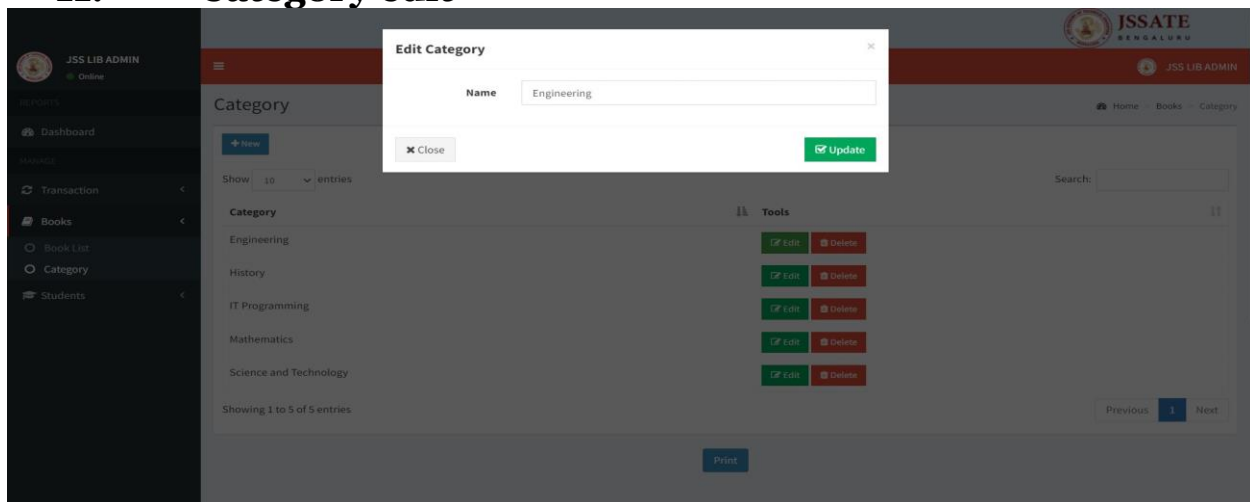
## 10. On clicking Books → Category



**Fig 8.13: Book Category page**

On clicking category option in Books the admin can view the list of book categories and add new book category as shown in the figure 8.13.

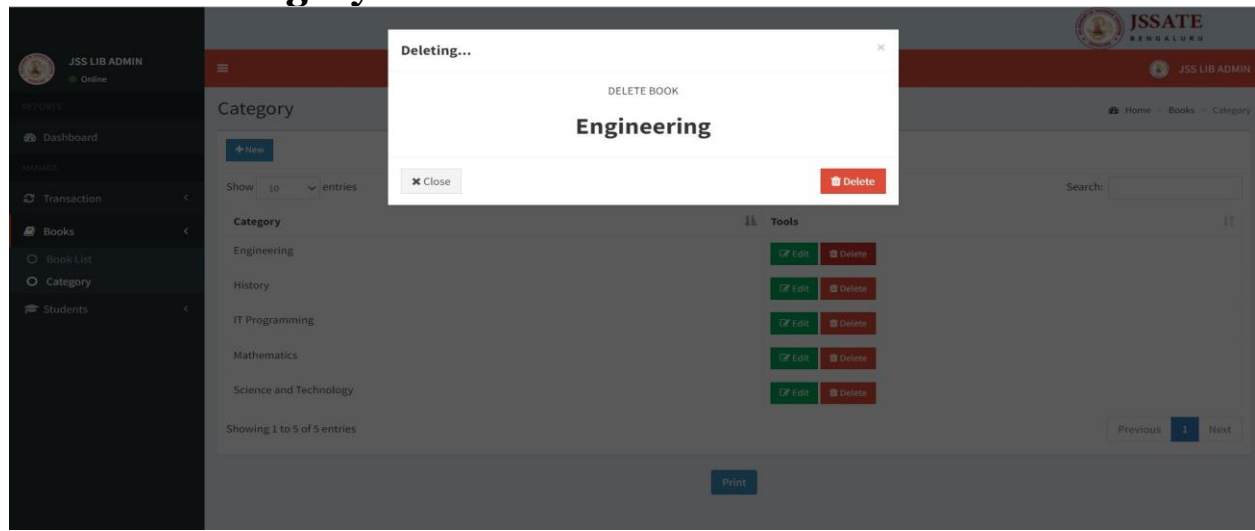
## 11. Category edit



**Fig 8.14: Category edit**

The category of the selected book can be updated as shown in the above figure 8.14.

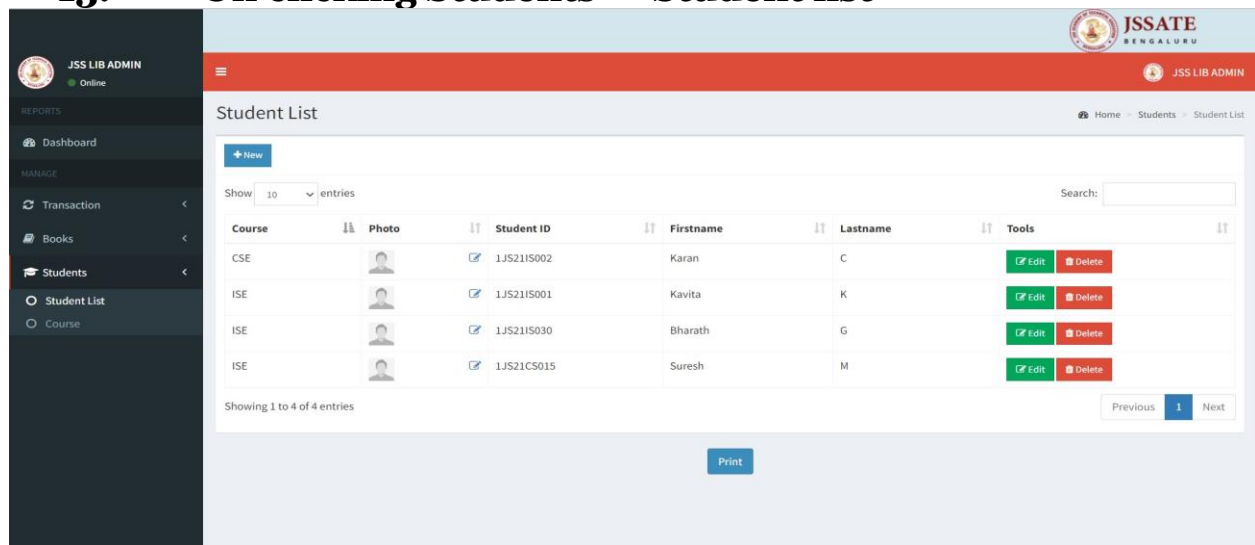
## 12. Category delete



**Fig 8.15: Category delete page**

The selected book's category can be deleted as shown in the figure 8.15.

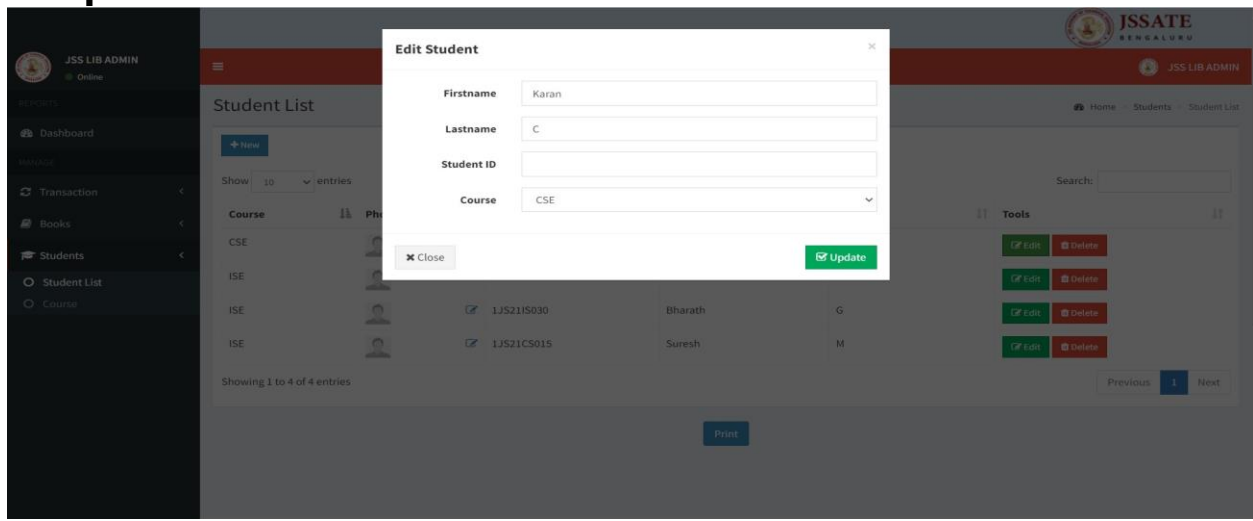
## 13. On clicking Students → Student list



**Fig 8.16: Student list**

On clicking Student list option in Students, the admin can view Student List and edit the Student details or delete student and add new student as shown in the above figure 8.16.

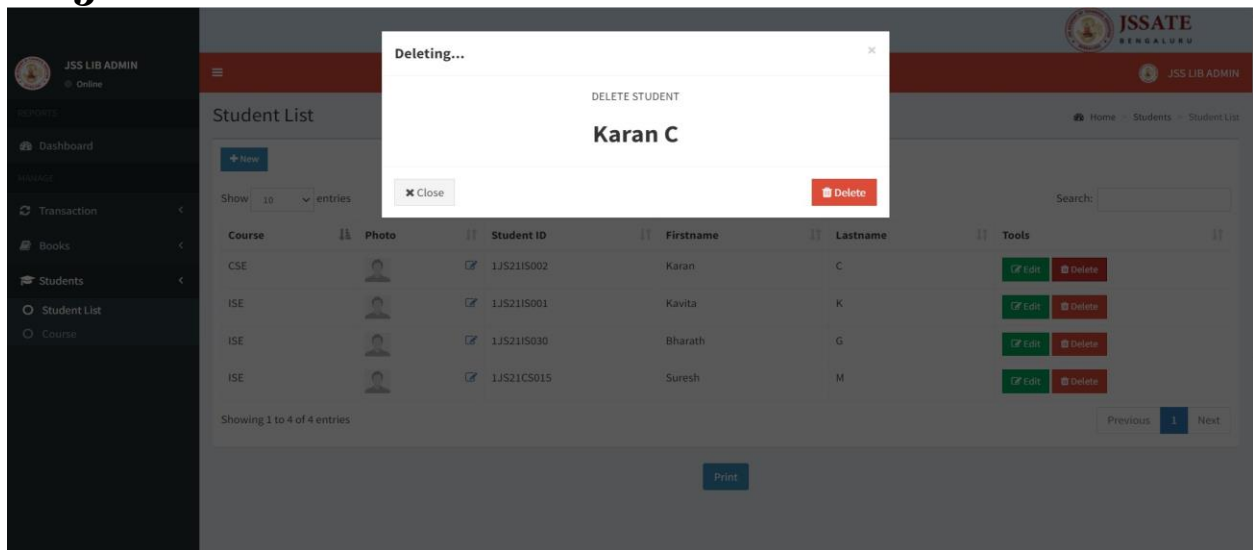
## 14. Edit student



**Fig 8.17: Edit Student page**

The various details like Student's First name, Last name, Student ID and Course of the book can be updated as shown in the figure 8.17.

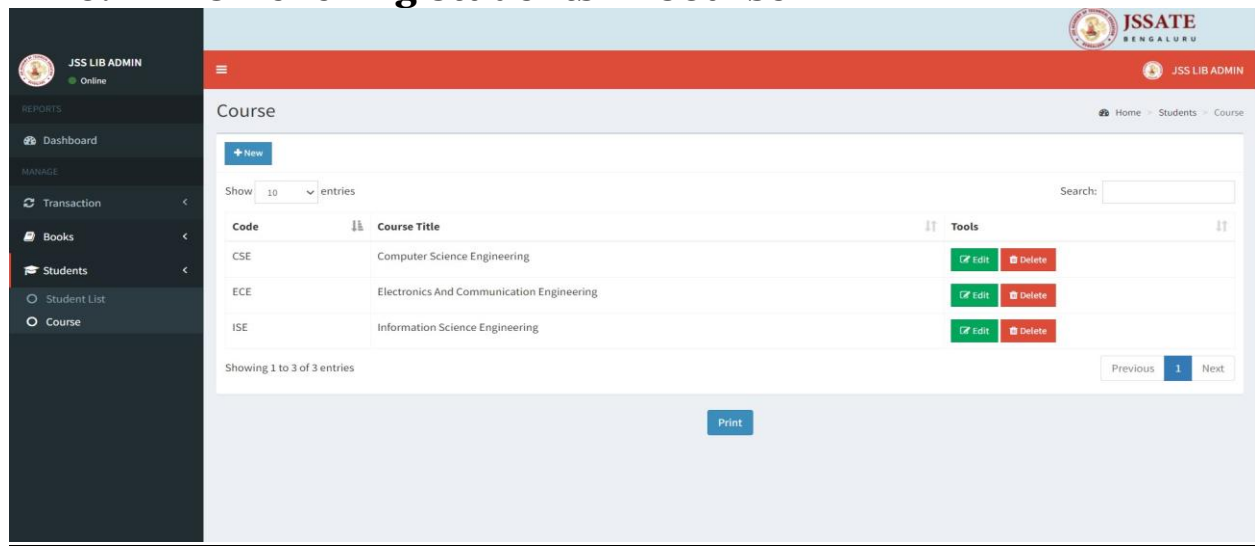
## 15. Delete student



**Fig 8.18: Delete Student page**

The selected student can be removed as shown in the figure 8.18.

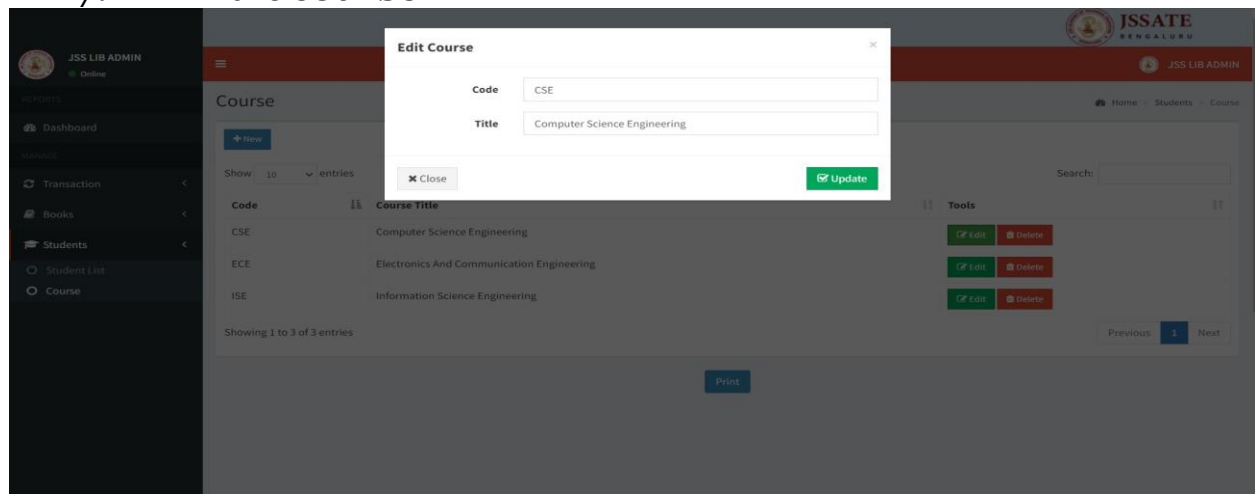
## 16. On clicking Students → Course



**Fig 8.19: Student course page**

On clicking course option in Students, the admin can view the list of courses, course titles and add new course as shown in the figure 8.19.

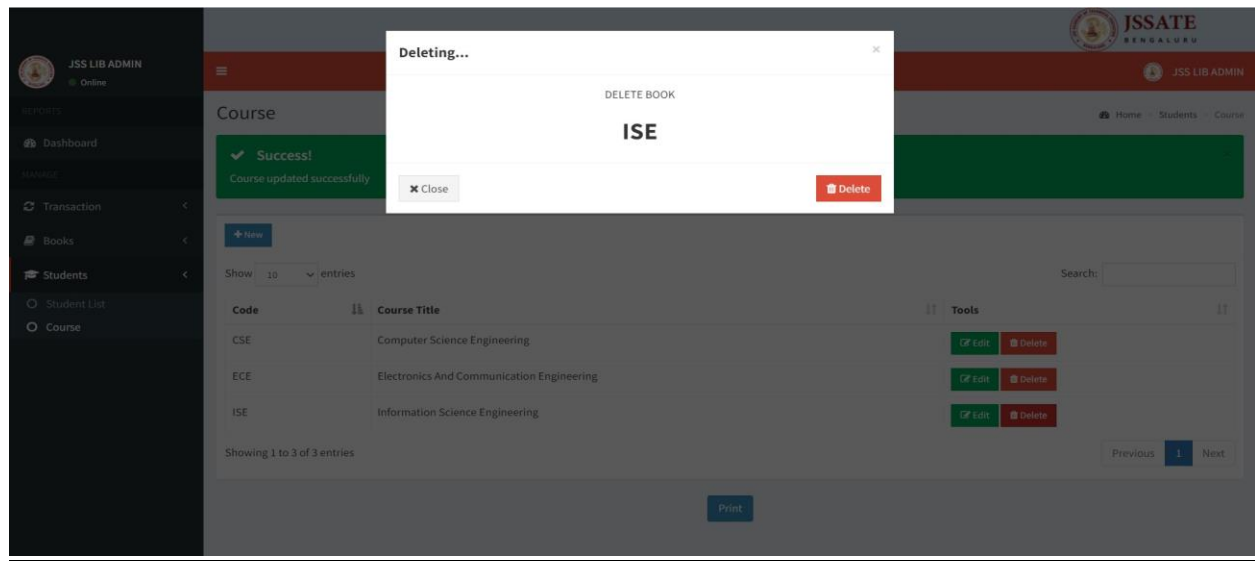
## 17. Edit course



**Fig 8.20: Edit course page**

The course of the selected course title can be updated as shown in the above figure 8.20.

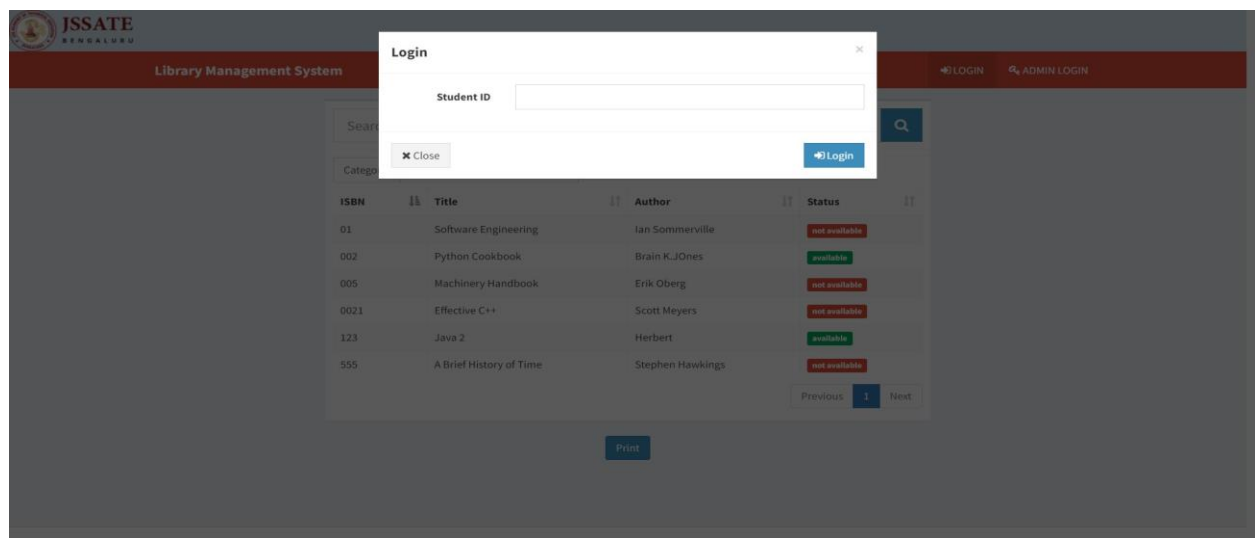
## 18. Delete course



**Fig 8.21: Delete course**

The selected course can be deleted as shown in the figure 8.21.

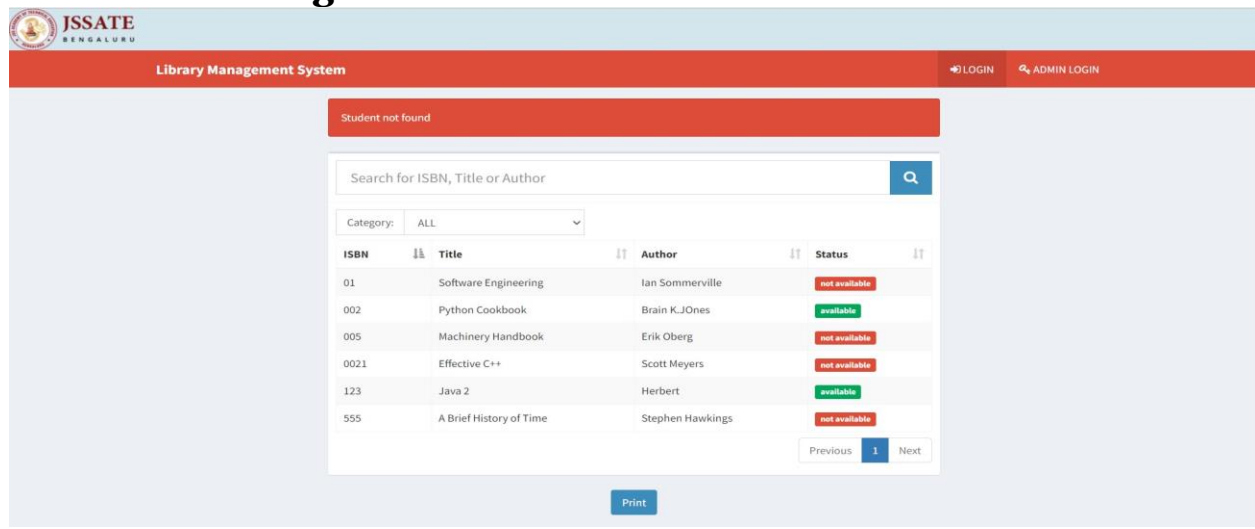
## STUDENT LOGIN



**Fig 8.22: Student login**

The above figure 8.22 is the login page for student. This page allows the student to gain access to Library Management System by entering student ID.

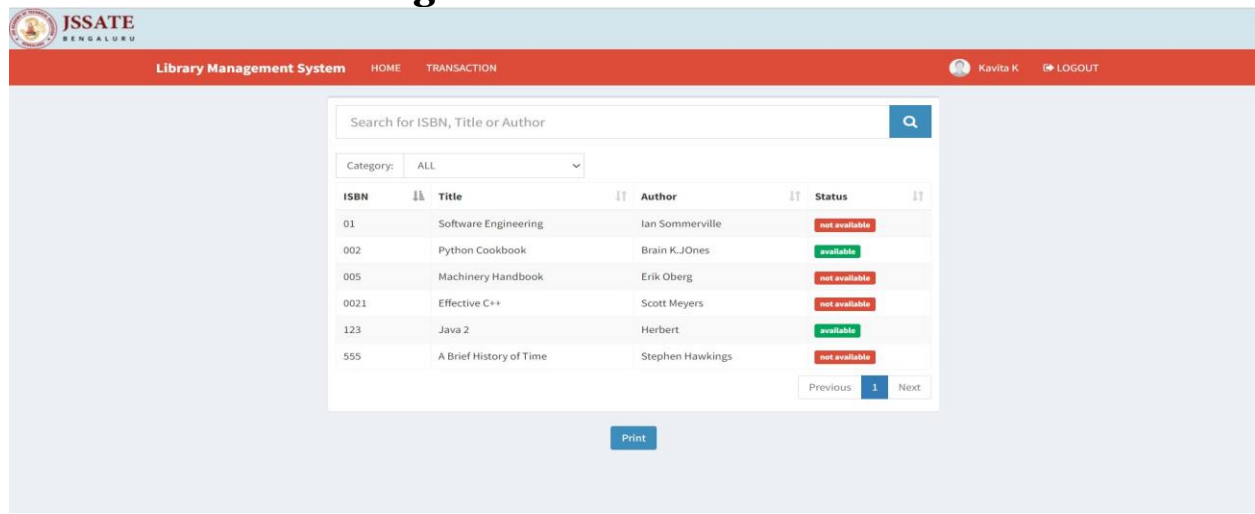
## 1. On entering incorrect student ID



**Fig 8.23: Error on entering incorrect Student ID**

The error notification is generated when the student enters the incorrect student ID as shown in the figure 8.23.

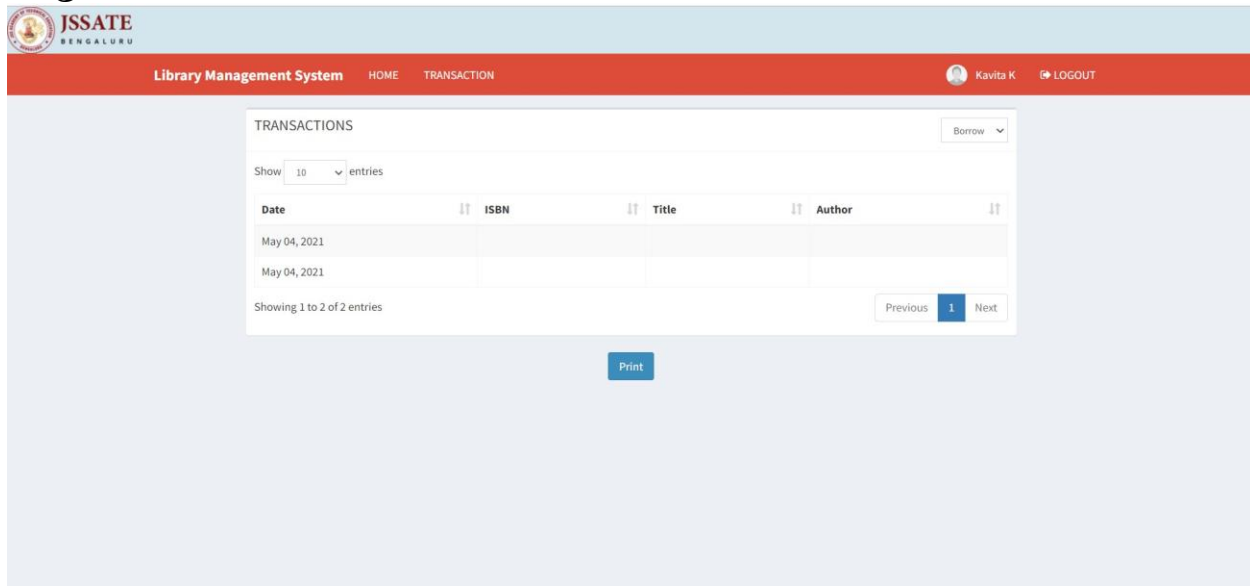
## 2. On successful login



**Fig 8.24: Successful Login**

The contents of Library Management System are displayed after the successful login by student as shown in the figure 8.24.

### 3. Transaction



The screenshot displays the 'TRANSACTIONS' page of the JSSATE Library Management System. The page header includes the JSSATE logo and the text 'JSSATE BENGALURU'. The navigation bar shows 'Library Management System', 'HOME', and 'TRANSACTION'. The user 'Kavita K' is logged in, and there is a 'LOGOUT' button. The main content area features a 'TRANSACTIONS' section with a 'Borrow' dropdown menu. Below this is a table with columns for 'Date', 'ISBN', 'Title', and 'Author'. The table shows two entries for 'May 04, 2021'. The page also includes a 'Show 10 entries' filter, a 'Showing 1 to 2 of 2 entries' indicator, and a 'Print' button.

Date	ISBN	Title	Author
May 04, 2021			
May 04, 2021			

**Fig 8.25: Transaction page**

The above figure 8.25 shows the book transaction details to the student.

## CHAPTER 9

# CONCLUSION AND FUTURE ENHANCEMENTS

---

### CONCLUSION

The project was developed to nurture the needs of faculty members concerning research paper data management. This easy to use computerized version of research paper data will not only help the faculty but will also help in ease of administration. In this entire process, the project ensures that data stored is easy to access and available at all times. The security and encapsulation of data are provided by the triggers, as the triggers also provide backup of the data.

This is a small prototype of a management application for research-based data in college. The limitation of the application is that it lacks enough features to be implemented in a real-life situation. Such an application, if built with professional expertise, can be a highly useful cost-effective way for universities to function efficiently. The whole project aims at easing and aiding universities and colleges to organize and retrieve the data with precision and help in saving time and effort

### FUTURE ENHANCEMENT

- Member management through multiple channels including website, email, whatsapp, messaging and other push notifications from mobile devices such as android and iPhone.
- Barcoding enables users to find the book's exact location and number of books available in real time and print library cards.
- Automated circulation and control - Librarians can send automatic email notifications and SMS alerts to students on overdue return of library materials on circulation.
- Mobile Library Landscape - Allow students to carry library everywhere they go and get real-time updates on their mobile devices. Students can conveniently access library collections from the classroom, campus and from anywhere.



## CHAPTER 10

### REFERENCES

---

#### BOOK REFERENCES

1. Database systems Models, Languages, Design and Application Programming, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. An Introduction to Database Systems” by Bipin Desai.
3. Database System Concepts” by Abraham Silberschatz and S Sudarshan.
4. “ Database Management Systems” by Raghu Ramakrishnan.

#### WEB REFERENCES

1. W3schools
2. <https://www.youtube.com>
3. <https://stackoverflow.com/questions>
4. <https://www.slideshare.net>
5. [www.tutorialspoint.com](http://www.tutorialspoint.com)
6. [www.javatpoint.com](http://www.javatpoint.com)
7. <https://www.php.net/docs.php>
8. <https://dev.mysql.com/doc/>