

# Library Management System

Mini Project Report -Database Lab (DSE 2260)

Department of Data Science & Computer Applications



B. Tech Data Science

4<sup>th</sup> Semester – B4

Submitted By

K Mani Chandra	200968212
Pranit Arunrao Jagtap	200968214
Gururi Ritheesh	200968218
Anwasha	200968222

**Mentored By**

Vinayak M  
Assistant Professor-Senior  
DSCA, MIT

Archana H/ Shameem  
Assistant Professor-Senior  
DSCA, MIT



**MANIPAL INSTITUTE OF TECHNOLOGY**  
**MANIPAL**  
(A constituent unit of MAHE, Manipal)

Date:

## **CERTIFICATE**

This is to certify that the K Mani Chandra (200968212), Pranit Arunrao Jagtap (200968214), Gururi Ritheesh(200968218), Anwasha(200968222) have successfully executed a mini project titled Library Management System rightly brining fore the competencies and skill sets they have gained during the course- Database Lab (DSE 2262 & DSE ), thereby resulting in the culmination of this project.

**Vinayak M**  
**Assistant Professor-Senior**  
**DSCA, MIT**

**Archana H / Shameem**  
**Assistant Professor-Senior**  
**DSCA, MIT**

## **ABSTRACT**

Online Library Management System is a system which maintains the information about the books present in the library, their authors, the members of library to whom books are issued, library staff and all.

This is very difficult to organize manually. Maintenance of all this information manually is a very complex task.

Owing to the advancement of technology, organization of an Online Library becomes much simple. The Online Library Management has been designed to computerize and automate the operations performed over the information about the members, book issues and returns and all other operations.

This computerization of library helps in many instances of its maintenances. It reduces the workload of management as most of the manual work done is reduced

# Contents

<b>1. Introduction</b>	<b>1</b>
<b>2. Synopsis</b>	<b>2</b>
<b>2.1 Proposed System</b>	<b>2</b>
<b>2.2 Objectives</b>	<b>2</b>
<b>3. Functional Requirements</b>	<b>3</b>
<b>4. Detailed Design</b>	<b>7</b>
<b>4.1 ER Diagram</b>	<b>9</b>
<b>4.2 Schema Diagram</b>	<b>10</b>
<b>4.3 Data Dictionary</b>	<b>12</b>
<b>4.4 Relational Model Implementation</b>	<b>13</b>
<b>4.5 Queries</b>	<b>14</b>
<b>4.7 Triggers</b>	<b>..</b>
<b>4.8 Stored Procedures</b>	<b>..</b>
<b>4.9 Stored Functions</b>	
<b>5. Implementation Functional Requirements</b>	
<del><b>6. Testing</b></del> No need to write testing	
<b>7. Result</b>	
<b>8. Conclusion and Future Work</b>	<b>20</b>
<del><b>References</b></del> not required	

# Chapter 1

## Introduction

## Introduction

A library is a collection of books which is made accessible to a student community for borrowing or reference sake. The collection of the resources and information are provided in physical format in a building. The main aim of this system is to develop a new programmed system that will conveying ever lasting solution to the manual base operations and to make available a channel through which staff can maintain the record easily and customers can access the information about the library at whatever place they might find themselves.

Library Management System allows the staff to store the book details and the borrower details. The system is strong enough to withstand regressive yearly operations under conditions where the database is maintained and cleared over a certain time of span. The implementation of the system in the organization will considerably reduce data entry, time and also provide readily calculated reports.

# Chapter 2

## Synopsis

### **2.1 Proposed System**

It keeps track of all the information about the books in the library, their cost, status and total number of books available in the Library. The user will find it easy in this automated system rather than using the manual writing system. The system contains a database where all the information will be stored safely.

### **2.2 Objectives**

The main objective of the library management system is to manage the details of the students, books, staff, borrower, shelf and categories. It is totally built at the administrative end and thus only the administrator is guaranteed access. The purpose of the project is to reduce manual work involved in managing all the details.

Features: - integration of all student records.

-keep track of all borrowing transactions.

-increase efficiency of managing all the details

## **Chapter 3**

### **Functional Requirements**

#### **3.1. REGISTER NEW USER**

Description of feature

This feature can be performed by all users to register new user to create account.

##### **Functional requirements**

- System must be able to verify information.
- System must be able to delete information if information is wrong.

#### **3.1.2 login**

Description of feature

This feature can be performed by only registered staff by their id to enter the system.

##### **Functional requirements**

- System must be able to identify the staff by his registration id.
- System must give access to the information if the registration id and password matches and reject if its wrong.

#### **3.2 REGISTER NEW BOOK**

Description of feature

This feature allows to add new books to the library

##### **Functional requirements**

- System must be able to verify information
- System must be able to enter number of copies into table.
- System must be able to not allow two books having same book id.

### **3.3 SEARCH BOOK BY DESCRIPTION OF FEATURE**

This feature is found in book maintenance part.

we can search book based on book id, book name, publication or by author name.

#### **Functional requirements**

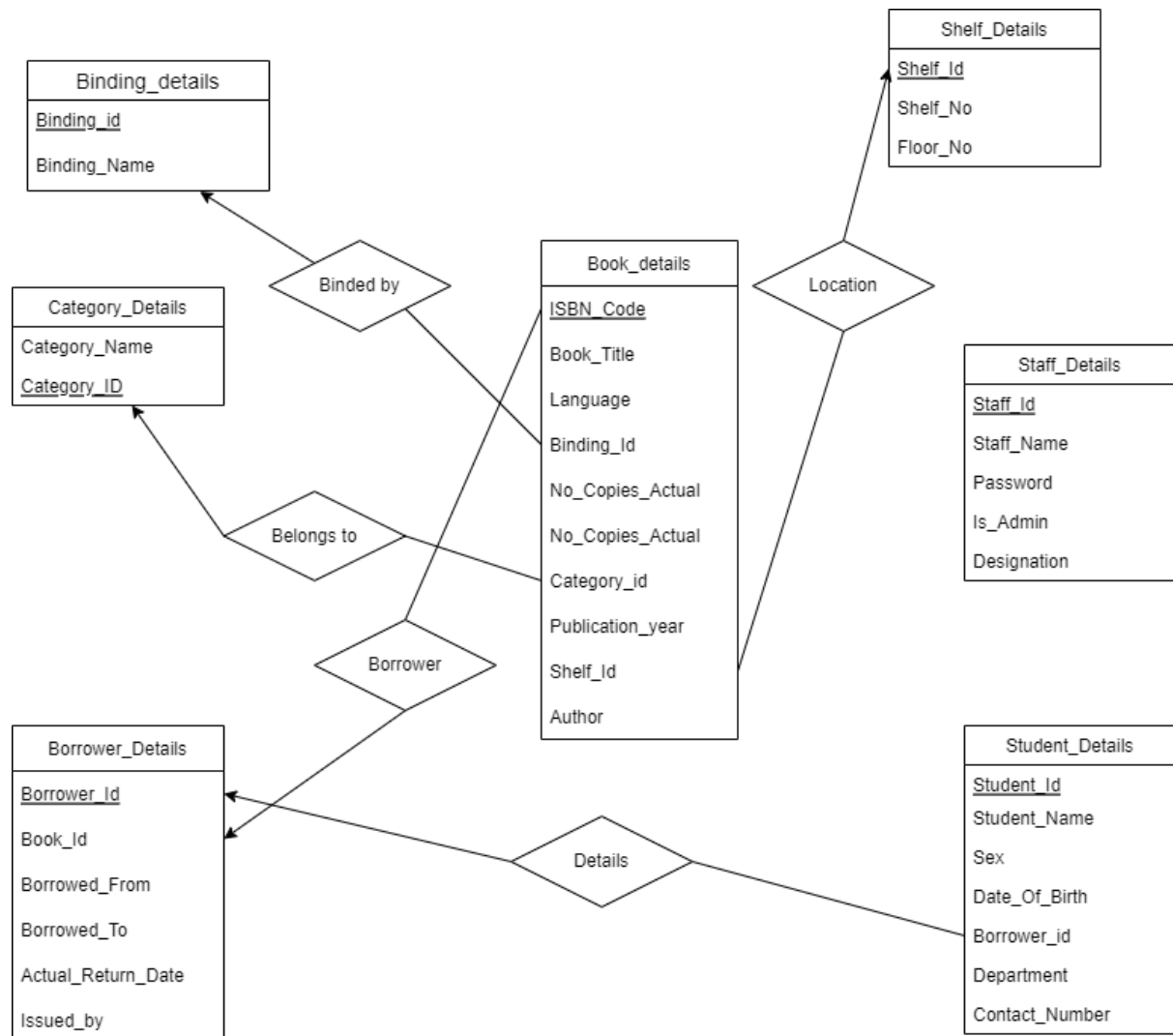
- System must be able to search the database based on select search type.
- System must be able to filter book based on keyword entered.
- System must be able to show the filtered book in table view Functional requirements.
- System should be able to add detailed information about events.
- System should be able to display information on notice board available in the homepage of site.



## Chapter 4

### Detailed Design

#### 4.1 ER Diagram



#### 4.2 Schema Diagram

**Book Details**(ISBN\_Code, Book\_Title, Language, Binding\_Id, No\_Copies\_Actual, Category\_Id, Publication\_Year, Shelf\_Id, Author)

Binding\_ID REFERENCES Binding\_Details;

Category\_Id REFERENCES Category\_Details,

Shelf\_Id REFERENCES Shelf\_Details

);

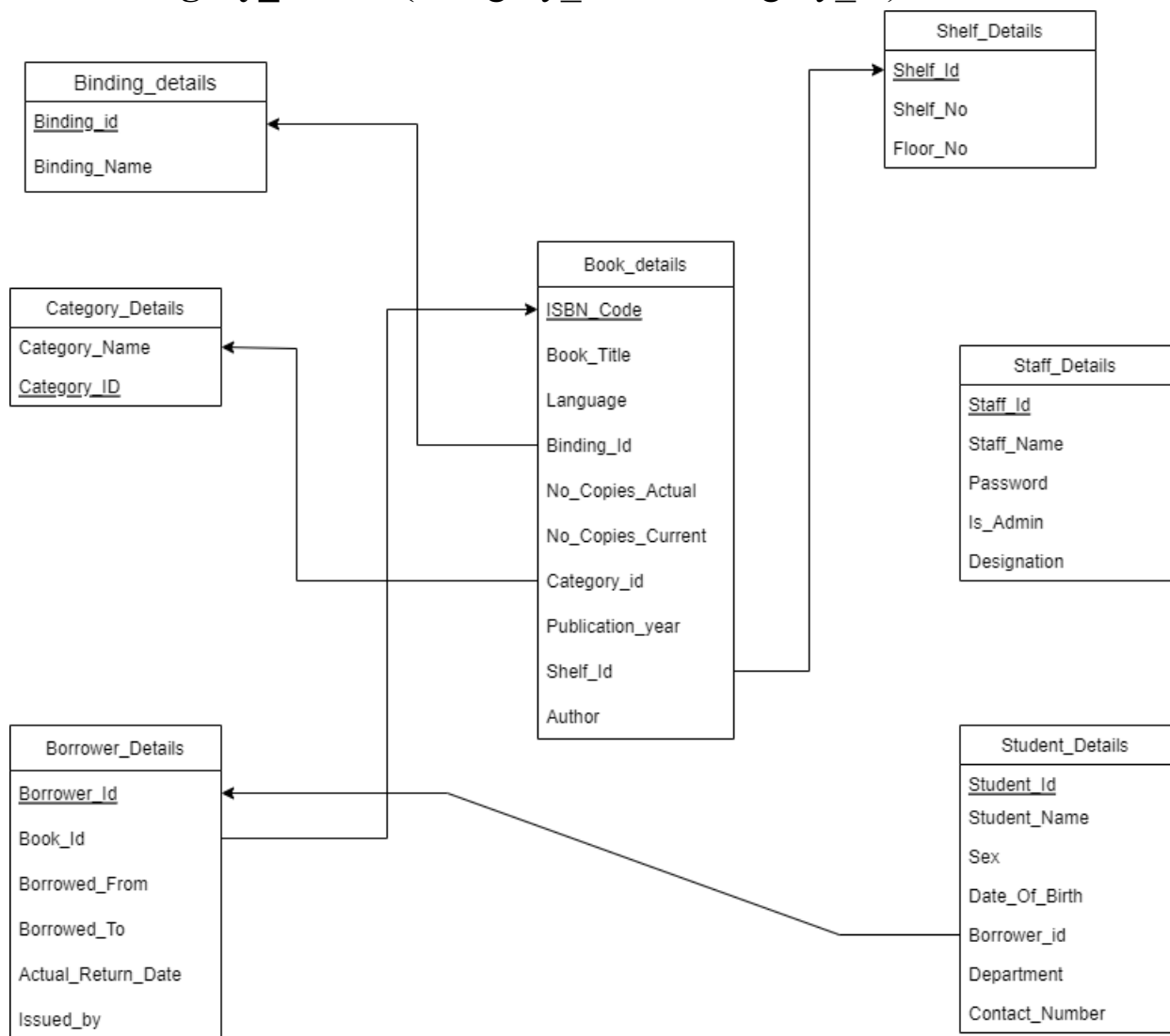
**Borrower\_details**(borrower\_id,Book\_id,  
Borrowed\_From,Borrowed\_to,Actual\_Return\_Date,Issued\_by)  
Book\_Id References Book\_details(ISBN\_code),  
Issued\_by REFERENCES Shelf\_Details

**Student\_Details** (Student\_Id ,Student\_Name ,Sex Date\_Of\_Birth  
,Borrower\_id ,Department, Contact\_Number)  
borrower\_id\_FK REFERENCES Borrower\_Details,

**Shelf\_details**(Shelf\_id, Shelf\_no, Floor\_no)

**Binding\_detials** (Binding\_id, Binding\_name)

**Category\_details** (Category\_name, Category\_id)



## 4.3 Data Dictionary

### BINDING\_DETAILS

column	Data type(size)	constraint	Constraint name
Binding_id	Number(12)	Primary key	Binding_details_pk
Binding_name	Varchar2(20)	-	-

### CATEGORY\_DETAILS

column	Data type(size)	constraint	Constraint name
Category_name	Varchar2(20)	-	-
Category_id	Number(12)	Primary key	Category_details_pk

### SHELF\_DETAILS

column	Data type(size)	constraint	Constraint name
Shelf_id	Number(5)	Primary key	Shelf_id_pk
Shelf_no	Number(3)	-	-
Floor_no	Number(2)	-	-

### STAFF\_DETAILS

column	Data type(size)	constraint	Constraint name
Staff_id	Number(10)	Primary key	Staff_details_pk
Staff_name	Varchar2(20)	-	-
password	Varchar2(20)	-	-
Is_admin	Binary_float	-	-
designation	Varchar2(20)	-	-

### STUDENT\_DETAILS

column	Data type(size)	constraint	Constraint name
Student_id	Number(20)	Primary key,foreign key references borrower_details	Student_id_pk, Borrower_id_FK
Student_name	Varchar2(20)	-	-
Sex	Varchar2(8)	-	-
Date_of_birth	date	-	-
Borrower_id	Number(20)	-	-
Department	Varchar2(20)	-	-
Contact_number	Number(20)	-	-

## BOOK\_DETAILS

column	Data type(size)	constraint	Constraint name
Isbn_code	Number(12)	Primary key,	Isbn_code_fk
Book_title	Varchar2(20)	-	-
language	Varchar2(20)	-	-
Binding_id	Number(12)	Foreign key references binding details	Binding_id_fk
Publication_year	Number(4)	-	-
Shelf_id	Number(5)	Foreign key references category_details	Shelf_id_fk
No_copies_actual	Number(3)	-	-
No_copies_current	Number(3)	-	-
Category_id	Number(12)	Foreign key references shelf_details	Category_id_fk

## BORROWER\_DETAILS

column	Data type(size)	constraint	Constraint name
Borrower_id	Number(12)	Primary key	Borrower_details_pk
Book_id	Number(12)	Foreign key references Book_details(isbn_code)	Book_id_fk
Borrowed_from	Varchar2(20)	-	-
Borrowed_to	Varchar2(20)	-	-
Actual_return_date	date	-	-
Issued_by	Varchar2(20)	Foreign key references staff_id	Issued_by_fk

## 4.4 Relational Model Implementation

```
CREATE TABLE Binding_details(  
    Binding_id number(12) PRIMARY KEY,  
    Binding_Name varchar2(20)  
);  
  
CREATE TABLE Category_Details (  
    Category_Name varchar2(20) ,  
    Category_Id number(12) PRIMARY KEY
```

```

);

CREATE TABLE Borrower_Details (
    Borrower_Id number(12) PRIMARY KEY,
    Book_Id number(12) CONSTRAINT Book_Id_FK References Book_details(ISBN_code),
    Borrowed_From date,
    Borrowed_TO date ,
    Actual_Return_Date date,
    Issued_by number(10) CONSTRAINT Issued_by_FK REFERENCES
    Staff_Details(Staff_Id);
);

CREATE TABLE Shelf_Details (
    Shelf_id number(5) PRIMARY KEY,
    Shelf_No number(3),
    Floor_No number(2)
);

CREATE TABLE book_details (
    ISBN_Code number(12) PRIMARY KEY
    Book_Title varchar2(20),
    Language varchar2(10),
    Binding_Id number(12) CONSTRAINT Binding_ID_FK REFERENCES Binding_Details;
    No_Copies_Actual number(3),
    No_Copies_Current number(3),
    Category_id number(12) CONSTRAINT Category_Id_FK REFERENCES Category_Details,
    Publication_year number(4),
    Shelf_Id number(5) CONSTRAINT Shelf_Id_FK REFERENCES Shelf_Details,
    Author varchar(20)
);

CREATE TABLE Staff_Details (
    Staff_Id number(12) PRIMARY KEY,
    Staff_Name varchar2(20),
    Password varchar2(20),

```

```

Is_Admin binary_float,
Designation varchar2(20)
);
CREATE TABLE Student_Details (
Student_Id number(20) PRIMARY KEY,
Student_Name varchar2(20),
Sex varchar2(8),
Date_Of_Birth date,
Borrower_Id number(12) CONSTRAINT borrower_id_FK REFERENCES
Borrower_Details,
Department varchar2(20),
contact_Number number(10),
);

```

## 4.5 Queries

.....

## 4.7 Triggers

**[ if applicable/ if implemented then show the code for Triggers]**

....

## 4.8 Stored Procedures

**[ if applicable/ if implemented then show the code for Triggers]**

...

## 4.9 Stored Functions

**[ if applicable/ if implemented then show the code for Triggers]**

.....

## **5. Functional Requirement Implementation**

[ code related to different function implementation may be added here]

## **6. Testing**

~~[-{set of testcases passed/failed}]~~

[No need to write these]

## **7. Result**

[screenshots of applications with 1-2-line explanation]

## **8. Conclusion and Future Work**

### **8.1 Conclusion**

....

### **8.2 Scope for future work**

....

## **References**