A Project Report on

Digital Library Management System

Submitted to Manipal University, Jaipur

Computers Science and Engineering

with Specialization in Artificial

Intelligence and Machine Learning

2021-2025

By

Akash Kumar Singh

219310279

MANIPAL UNIVERSITY
JAIPUR

Under the guidance of

Dr. Rishi Gupta

Department of Artificial Intelligence and Machine Learning

School of Computer Science and Engineering

Manipal University Jaipur

Jaipur, Rajasthan

Introduction

The Database Library System is intended to Automate the borrowing of books by students. First all the students will have to create an account on the platform using their university's email address and then verify the same via an otp. Users trying to borrow books just have to fill up some info on the application which will be confirmed while going out of the library using a sensor. While returning the book if the return date is greater than the date when it should have been returned then a fine is added to the student's account. Library activities such as creating a new borrower, giving books to the borrowers, maintaining the details of allthe item that were available in the books. This also helps the librarians by providing information such as total copies available of each book etc.

Motivation

Library Management System stores every information electronically and in an organized and systematic way which leads to effective results. It enhances the overall performance of the students and develops the habit of silent reading. The software is designed in such a way that it modernizes the library management system and help the students to make the best use of Library Automation System. It saves the students from lining up to borrow books and directly issue the book from the application. It also helps the student to search for a particular book and check whether it is currently available in the Library or not.

Project Objective

The project aims to create an alterable database to increase efficiency of Library Management system. This project aims to make the borrowing of books easier and automated. Even so,this system has the following advantages and disadvantages:

Advantages of Online Library Management System are:

1. It is user-friendly software.
2. It is cost-effective and easy to install.
3. It increases the efficiency.
4. The data is secure as the database is on phpmyadmin.co

Disadvantages of Online Library Management System are:

1. Complicated to operate.
2. Online Systems require high-speed internet connectivity.

Methodology/ Planning of work:

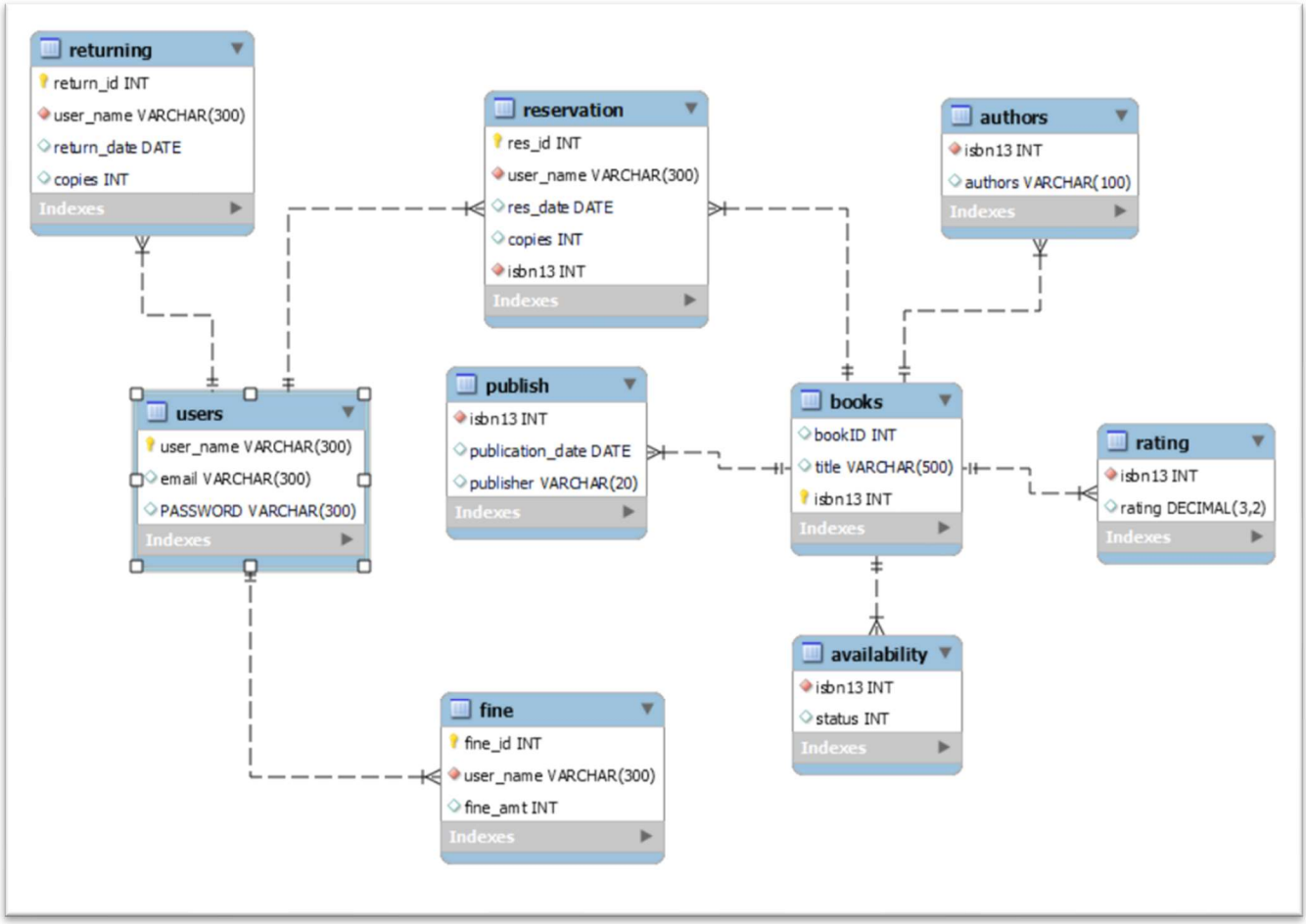
In order to tackle the problem, the work is divided into following phases:

1. **Reconnaissance:** This phase will help to identify the problem and ideate the project.
2. **Planning:** This phase will create an ER Diagram, Relational Schema and SQL Queries to generate tables for the database.
3. **App Launch:** Creation and Launching of desktop app.

Facilities required for proposed work:

1. MySQL Server
2. PHPMyAdmin
3. Python
4. Tkinter Library for GUI

Enhanced Entity Relationship Diagram:



Print View – Data dictionary:

AUTHORS

Column	Type	Null	Default	Comments
isbn13	int(11)	No		
authors	varchar(100)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
fk_authors_books	BTREE	No	No	isbn13	0	A	No	

AVAILABILITY

Column	Type	Null	Default	Comments
isbn13	int(11)	No		
status	int(1)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
fk_availability_books	BTREE	No	No	isbn13	0	A	No	

BOOKS

Column	Type	Null	Default	Comments
bookID	int(11)	Yes	*NULL*	
title	varchar(500)	Yes	*NULL*	
isbn13 *(Primary)*	int(11)	No	0	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	isbn13	0	A	No	

FINE

Column	Type	Null	Default	Comments
fine_id *(Primary)*	int(11)	No		
user_name	varchar(300)	No		
fine_amt	int(11)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	fine_id	0	A	No	
fk_fine_users	BTREE	No	No	user_name	0	A	No	

PUBLISH

Column	Type	Null	Default	Comments
isbn13	int(11)	No		
publication_date	date	Yes	*NULL*	
publisher	varchar(20)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
fk_publish_books	BTREE	No	No	isbn13	0	A	No	

RATING

Column	Type	Null	Default	Comments
isbn13	int(11)	No		
rating	decimal(3,2)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
fk_rating_books	BTREE	No	No	isbn13	0	A	No	

RESERVATION

Column	Type	Null	Default	Comments
res_id *(Primary)*	int(11)	No		
user_name	varchar(300)	No		
res_date	date	Yes	*NULL*	
copies	int(11)	Yes	*NULL*	
isbn13	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	res_id	0	A	No	
fk_reservation_users	BTREE	No	No	user_name	0	A	No	
fk_reservation_books	BTREE	No	No	isbn13	0	A	No	

RETURNING

Column	Type	Null	Default	Comments
return_id *(Primary)*	int(11)	No		
user_name	varchar(300)	No		
return_date	date	Yes	*NULL*	
copies	int(11)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	return_id	0	A	No	
fk_returning_users	BTREE	No	No	user_name	0	A	No	

USERS

Column	Type	Null	Default	Comments

user_name *(Primary)*	varchar(300)	No		
email	varchar(300)	Yes	*NULL*	
PASSWORD	varchar(300)	Yes	*NULL*	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	user_name	0	A	No	

Tables In Database

1. Books:

This table contains the complete list of books that are available in the library. Each Book is provided with a unique ISBN which serves as a primary key. The book details include the bookID, ISBN13, Book Title.

bookID : This is just for facilitating counting the number of books in the library, and to quickly identify the book as we have a different id unique to books present in the library.

ISBN13: This is unique ID given to every book. Since there may be a large no. of books with same TITLE, this ISBN no. will help us to distinguish between booksof same title.

Title: Provides the name of the book.

2. Availablility:

This table contain the availability of the book.

Isbn13: Contains the book's isbn13 number.

Status: If the book is currently available or not(0-> Not Available, 1->Available).

3. Authors:

This table contains the isbn13 and authors of the book.

4. Fine:

This table contains fine id, username of the person with the fine and the fine amount.

5. Publish:

This table contains the isbn13, publication date of the book and publisher name.

6. Rating:

This table contains the isbn13, rating of the book.

7. Reservation:

This table contains the reservation_id for the reservation, user_name of the person borrowing the book, reservation date, number of copies being reserved and isbn13 of the book being borrowed.

8. Returning:

This table contains the return id, user_name of the person returning the book, return date and copies returned.

9. Users:

This table contains the user_name, email and password for the person making an account to access the digital library..

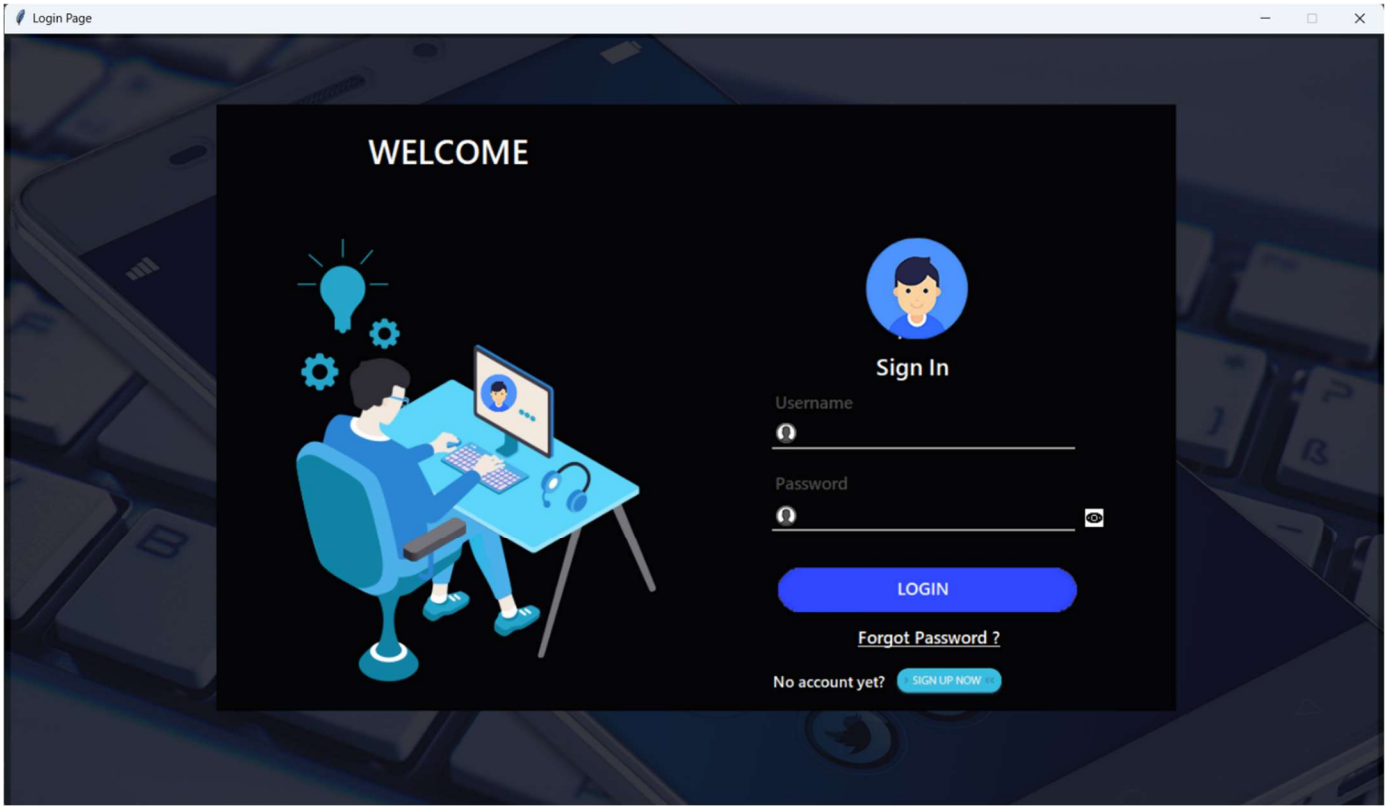
Library Management System (SQL Commands)

```
CREATE TABLE USERS (
user_name varchar(300),
email varchar(300),
PASSWORD varchar(300),
CONSTRAINT pk_users PRIMARY KEY(user_name)
);

CREATE TABLE BOOKS (
bookID integer,
title varchar(500),
isbn13 integer,
CONSTRAINT pk_books PRIMARY KEY(isbn13)
);

CREATE TABLE AUTHORS (
isbn13 integer not NULL,
authors varchar(100),
CONSTRAINT fk_authors_books foreign key(isbn13) references BOOKS(isbn13)
);

CREATE TABLE RATING(
isbn13 integer not NULL,
rating decimal(3,2),
CONSTRAINT fk_rating_books foreign key(isbn13) references BOOKS(isbn13)
);

CREATE TABLE PUBLISH(
isbn13 integer not NULL,
publication_date date,
publisher varchar(20),
CONSTRAINT fk_publish_books foreign key(isbn13) references BOOKS(isbn13)
);

CREATE TABLE AVAILABILITY(
isbn13 integer not NULL,
status integer(1) check (status = 0 OR status = 1),
CONSTRAINT fk_availability_books foreign key(isbn13) references BOOKS(isbn13)
);

CREATE TABLE RESERVATION(
res_id integer not NULL,
user_name varchar(300) not NULL,
res_date date,
copies integer,
isbn13 integer not NULL,
CONSTRAINT pk_reservation primary key(res_id),
CONSTRAINT fk_reservation_users foreign key(user_name) references USERS(user_name),
CONSTRAINT fk_reservation_books foreign key(isbn13) references BOOKS(isbn13)
);
```

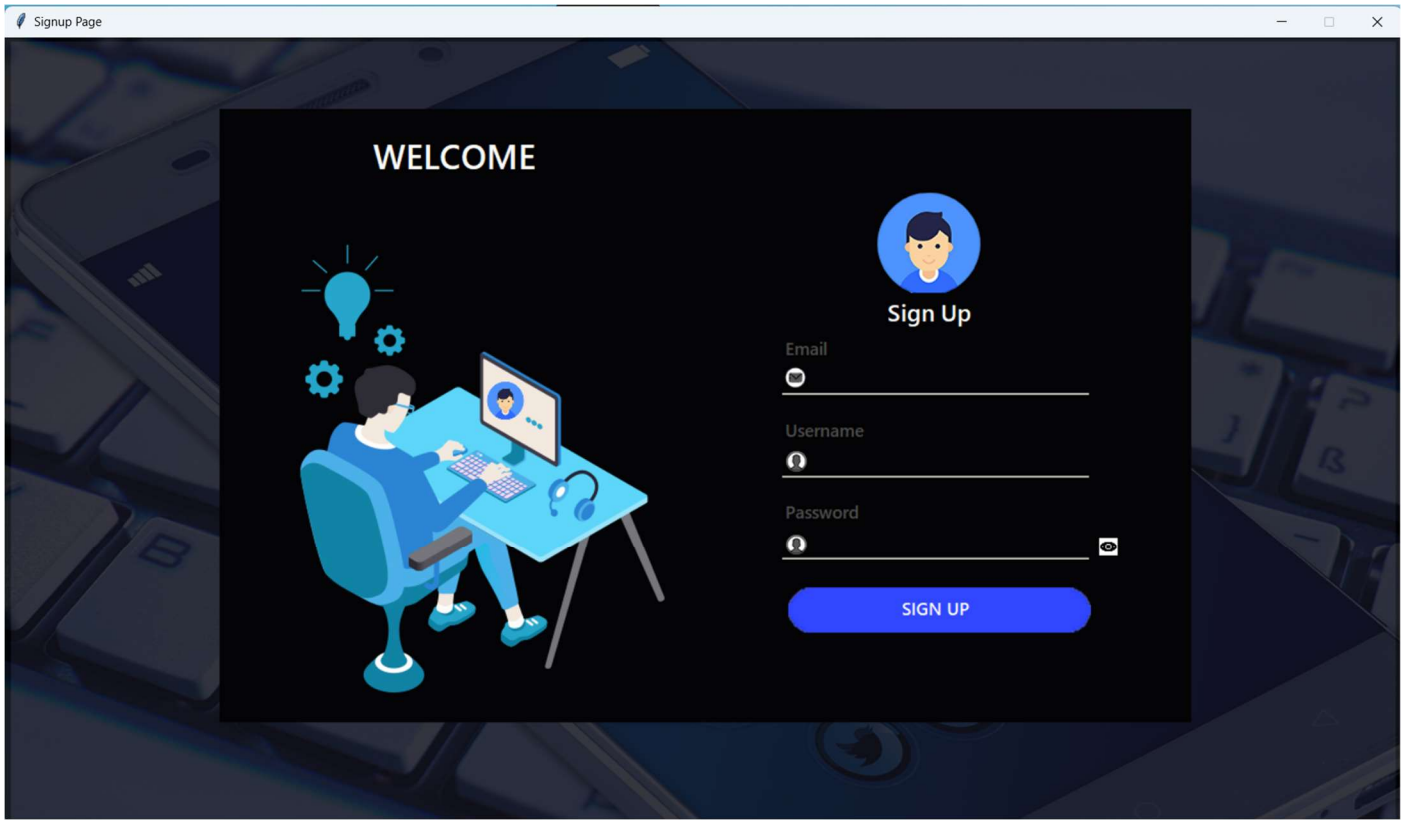
```
CREATE TABLE RETURNING(
return_id integer not NULL,
user_name varchar(300) not NULL,
return_date date,
copies integer,
CONSTRAINT pk_returning primary key(return_id),
CONSTRAINT fk_returning_users foreign key(user_name) references USERS(user_name)
);


CREATE TABLE FINE(
fine_id integer not NULL,
user_name varchar(300) not NULL,
fine_amt integer,
CONSTRAINT pk_fine primary key(fine_id),
CONSTRAINT fk_fine_users foreign key(user_name) references USERS(user_name)
);
```

Desktop Application Look:

Login Page:



Sign Up Page


Signup Page
WELCOME
Sign Up
Email
Username
Password
SIGN UP

CONCLUSION:

The database will help keep track of all the transactions that take place. It will also be holding information of all the books and students in order to maintain a history and contact points. I would like to thank Dr. Rishi Gupta for his teachings which have helped me to make this project.

Bibliography/References:

1. Oracle Academy
2. Database System Concepts by Abraham Silberschatz, Henry F. Korth, S. Sudarshan
3. phpMyAdmin
4. stackoverflow