Project Management Report

Library Management System

Disclaimers

The information contained in this document is the proprietary and exclusive propertyof Nimisha except as otherwise indicated. No part of this document, in whole or in part,may be reproduced, stored, transmitted, or used for design purposes without the prior written permission of Nimisha.

The information contained in this document is subject to change without notice.

The information in this document is provided for informational purposes only. Nimisha specifically disclaims all warranties, express or limited, including, but not limited, to the implied warranties of merchantability and fitness for a particular purpose, exceptas provided for in a separate software license agreement.

Privacy Information

This document may contain information of a sensitive nature. This information should not be given to persons other than those who are involved in the Library Management System project or who will become involved during the lifecycle

Document Owner

The primary contact for questions regarding this document is:

Author: Nimisha Doshi

Project Name: Library Management System

Email: 201900273@vupune.ac.in

1 Introduction

We created a Library Management Software just using a server-side database and Tkinter, which can do all sorts of things.

1.1 Scope

1. We'll create a simple home page, which will give the users the choice of whether they want to login as an **employee** (with all the powers that authorities have), or as a **student** (with fewer options).

2. We'll create a registration page for first time users, and also a login page for existing users. (For employees as well as students).

3. Once logged in, the user will get certain menu choices. If logged in as an employee they have all the authority to add a book, search for books, delete a book or even issue any book to the students. But if they're logged in as a student they only get two options; to either view all the books or to search for a particular one.

a. References

https://github.com/S-ayanide/Library-Manager

b. Methodology, Tools, and Techniques

Home Page

This will be the main dashboard whenever anyone opens our application. It gives the user two choices; to either log in as an employee or as a student.

Registration and Login

Upon user click, we'll have separate windows for employee and student registration and login. (With different attributes).

Let's talk about these simple functions one by one.

Add books

A window where employees can add books into the library with certain information regarding the book. (Book ID, name, etc)

Delete book

A window where employees can delete their books from the library.

View all books

This window shows us a tabular form of all the books present in our library. Whenever we add any new book that shows up here, any changes we make in the table will be reflected here.

Search book

It is exactly similar to **delete book.** The only difference is that it globalizes the frames and deletes them after searching as it displays the result in the same frame.

Issue book

This window gives the option to the employee to issue a book to the student, provided that he enters the correct book ID and the correct roll number of the student, as well as the correct employee ID.

c. Points of Contact

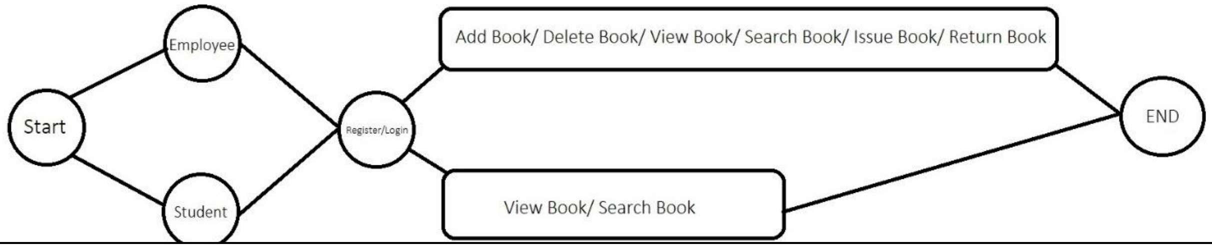
Nimisha Doshi

E-mail: 201900273@vupune.ac.in

1. Design Overview

Library Management System is a project made using Python and MySQL Database which will help to keep all the records of the library i.e., issued books, available books, etc.

Activity Network Diagram


Start
Employee
Student
Register/Login
Add Book/ Delete Book/ View Book/ Search Book/ Issue Book/ Return Book
View Book/ Search Book
END

a. Background Information

As we all know it is really difficult for any person to keep records of numbers of books, available books, issued books, returned books in the library on paper and registers which can get damaged or lost. So, to deal with the problems we have created a Library Management System which can help all the librarians to keep records of books which are available and issued in system.

b. System Evolution Description

This system is more efficient in the following ways -

1. The customer doesn’t now need to record on paper and registers he can directly save them into the database of the system.
2. The application is secure as we had implemented user authentication.
3. Overall process is fast.
4. The application is more robust.
5. Easy to use.

c. Technology Forecast

We have planned to implement 2 FA as an extra layer of security in the upcoming weeks as it will increase the security of the application.

d. Application Overview

As it is difficult to keep records of everything for a librarian on registers and papers. We have proposed this idea of a Library Management System which will help to keep all the records of Library i.e., issued books, available books, etc.

e. Current Process

In Library Management System we have implemented authentication for both employees and Students. A student can log in and search for the book and see its availability. An employee can add books or remove books, he/she can issue books to a student, return the book.

f. Proposed Process

As we know Two Factor Authentication (2FA) is a second layer of account protection, we have planned to implement in near future. We have planned to implement it using OTP (One Time Password) method.

g. Business Context

All the colleges, schools, private libraries, study halls can be the customers. Vishwakarma University can be considered a stakeholder in case finances are required.

h. Constraints

We have used Python 3 and Tkinter for graphical user interface and MySQL to store the database.

i. Risks

Identify the risks associated with the document, including contingencystrategies.

Risk	Low	Med.	High	Contingency
It is vulnerable SQL Injection				

Table 1 — Risks

j. Issues

List any outstanding issues that may affect the design document.

Ref	Issue	Action
1.	Not added count to same duplicate books.	Added different book id to each book.
2.	Books can be returned by any employee.	Added employee id if you wanted to return.
3		

Table 2 — Issues

k. Assumptions

List all assumptions regarding the design effort.

Re f	Assumption	Impact
1.	Students and Teacher can login.	No special login for management they have to login as Employee or Student.
2.		
3		

Table 3 — Assumptions

I. Dependencies

List the main dependencies regarding the design effort.

Re f	Dependency	Actio n
1.	Book.	Delete and Addition.
2.	Student's Name, Id, etc.	Search book.
3	Employee Name, Id, etc.	Search book, issue book, return book, add book and delete book.

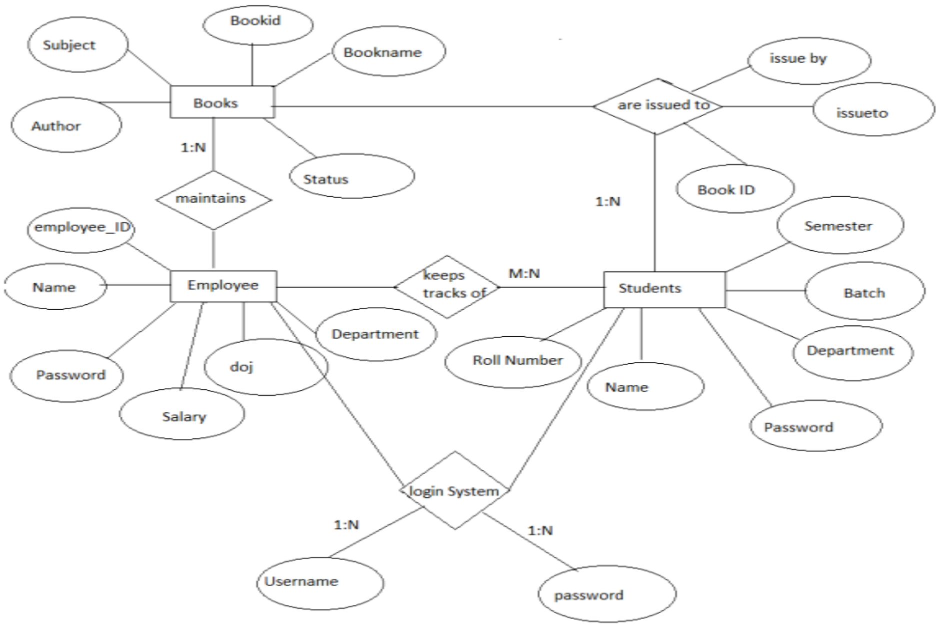
Table 4 — Dependencies

2. Scope of Work

In this chapter, describe the business and technical requirements that thecustomer has requested. Outline the scope of work, including the inputs, processing functionality, and outputs.

a. System-wide design decisions

- Book Entity: BookID, Book Name, Author, Subject, Status.
- Employee Entity: Emp_ID, Name, password, Salary, Date of joining (doj), Department.
- Student Entity: Batch, password, Name, Roll no, semester, department.
- Issue Entity: Issueid , Bookid , issueto, issued by.



b. System Functions

- Login system : login using username and password.
- Register system : for employee and student.
- If employee then he/she maintains the books and keeps track of books issued to students.

- Books : id is maintained along with its status if its issued to a student or avaliable.
- If Student: the record of student details and which books have been issued to each student is maintained.

c. Similar System Information

- The library software is developed using server side database and tkinter.
- The database used is SQL which is connected to library software to store student details, employee details and book details.
- The Tkinter gui is used for running the software on an online platform.

d. User Characteristics

- The employees and students can register and login through the system.
- The employee details are stores such as id, salary , doj etc and similarly student details are stored such as id, name , batch , division etc.
- The employee can add book details and view them if needed. He/ she can issue books to students and update the action in the soft
- The student can login to check which books are avaliable and books issued to them.

e. User Problem Statement

- Employees may not be able to keep track of books issued to students manually.
- The students have to be physically present to find out if the book is available or not.
- It becomes tedious task for employees too to search the book manually if its available or not.

f. User Objectives

- Security features
- Efficiency
- Availability
- Improved UI

g. Performance Requirements

Describe the performance requirements.

Faster GUI

Hardware :

Required storage : 20 mb

Compatible : windows, linux , mac

Software :

Python version : 3.9 and above

MySQL version : 8.0 and above

h. Security Requirements

Login validation system

i. Hardware Interfaces

PC / Laptop

Keyboard

Mouse

j. Communications Interfaces

Our software does not use any network interface.

k. Software Interfaces

Tkinter

IDE : Visual studio

l. Design Constraints

We have used Python 3 and Tkinter for graphical user interface
and MySQL to

store the database.

UI development required.

m. Data Analysis

Describe the data elements, characteristics, and their behavior values.

Book Entity :

Data Element	Characteristics	Behavior

Bookid	int	Stores the id of the books
Book name	String	Stores the name of the books
Author	String	Stores the name of the author
Subject	String	Stores the name of the subjects
Status	String	Shows if a book is available or not

Table 5 — Data Analysis

Employee Entity :

Data Element	Characteristics	Behavior
empid	int	Stores the id of employees
Name	String	Stores the name of the employees
Password	String	Stores the password of the respective employee
DOJ	Int	Stores the date of joining
Department	String	Stores department to which the employee belongs too.

Student entity :

Data Element	Characteristics	Behavior
Name	String	Stores the name of the students

Batch	String	Which batch the student belongs to
Password	String	Stores the password of the respective employee
Department	String	Stores department to which the student belongs to.
Roll no	Int	The roll no of the student
Semester	String	Which semester the students is in

Issue entity:

Data Element	Characteristics	Behavior
issueid	int	Stores the info about the issued bookid
Issueto	String	Stores the names of the students to whom the books are issued to
Issuedby	String	Stores the details of the employee who issued the book

n. Output Specifications

1. Details of Books : availability and author specifications.
2. Employee can view the student records.

o. Decision Tables

Outline the decision tables required to make decisions during proc-essing.

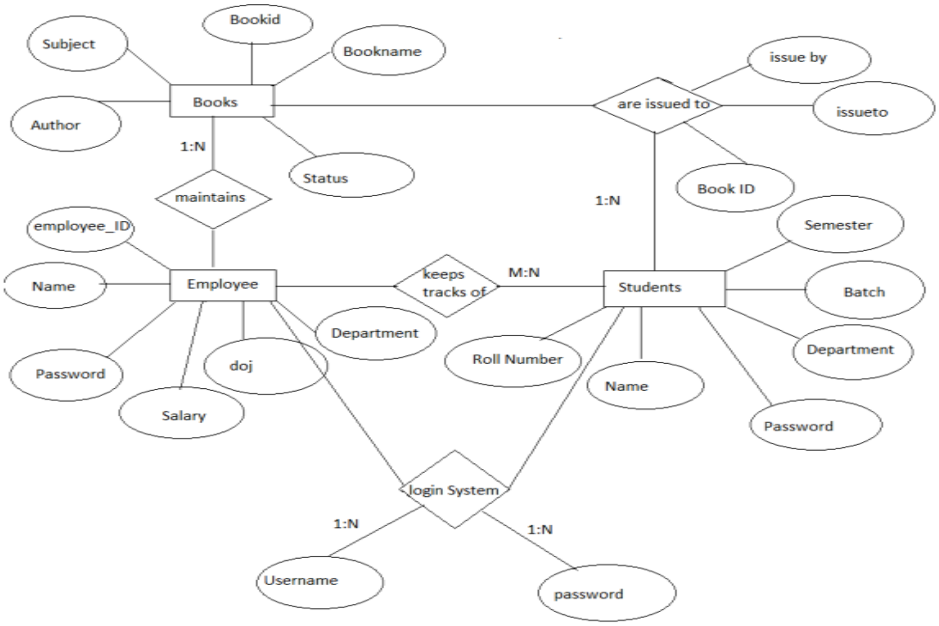
Table 6 — Decision Tables

Business condition	action
Valid selection	yes
New member	yes
Renewal	yes
Ask student details	yes
Ask employee details	yes
Issue books to students	yes
Login / logout	yes

p. Logical Datab

q. Logical Database Model

- Login system : login using username and password.
- Register system : for employee and student.
- If employee then he/she maintains the books and keeps track of books issued to students.
- Books : id is maintained along with its status if its issued to a student or avaliable.
- If Student: the record of student details and which books have been issued to each student is maintained.



r. Data Conversion

Once the user enters his/her details , the data gets stored in our sql database in its respective sections.

Once employee logs in the system :
add book details
Issue books to students
Delete a book from the database
View book list
Return a book

Once student logs in the system :
View book
Search book

s. Value Definitions

Describe the value of each unit of code in the system.

Field	Code	Value

Getting employee details	empdetail	EmpRegister()
		gettingEmpDetails()
		Employee()

Getting student details	studetail	studentRegister()
		gettingStuDetails()
		Student()

Getting book details	books	searchBook()
		ViewBooks()
		returnBook()
		addBooks()
		deleteBook()

Issue books	issuedetail	issueBooks()

Table 7— Value Definitions

t. External System Dependencies

Describe the dependencies the new system has on other [external] systems.

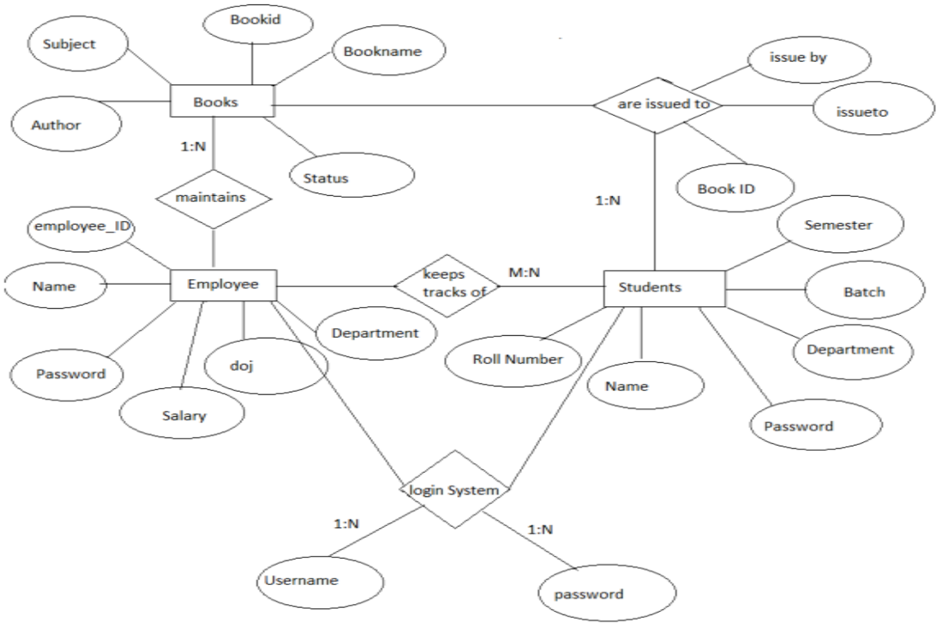
External System	Dependency
Tkinter	GUI
pymysql	Database software : to connect database with software
Pillow	Add image processing capabilities to our software

Table 8 — External System Dependencies

u. Data Validation

Secure login and validation system.

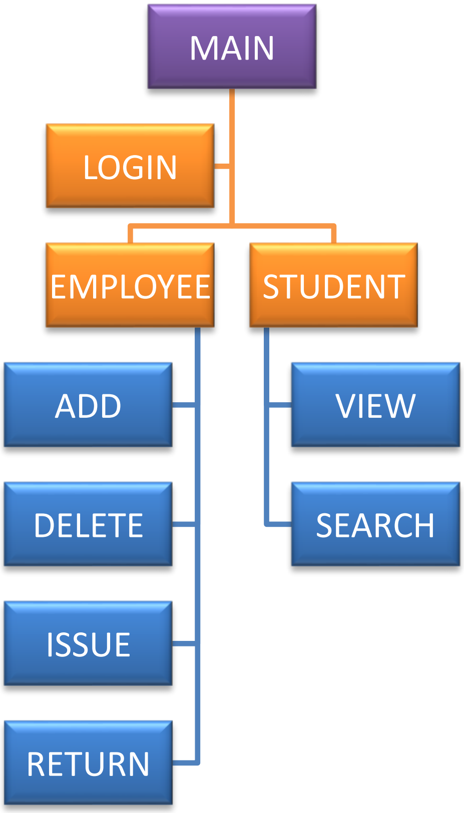
v. Data Migration and Transformation



- Manually access the student details and employee details in our database and manage by updating or removal of required data.

4 System Design

4.1 System Architechture



4.2 Modules and interaction

Modules used :

Tkinter for gui
Pillow : image - processing capabilities
Pymysql : connects our software to the database

4.3 Data Design

4.4 Database description

Database used : mysql

Internal Data Structure :

- Book Entity: BookID, Book Name, Author, Subject, Status.
- Employee Entity: Emp_ID, Name, password, Salary, Date of joining (doj), Department.
- Student Entity: Batch, password, Name, Roll no, semester,department.
- Issue Entity: Issueid , Bookid , issueto, issued by

Global Data Structure:

Books table :	book
Student details table :	studetail
Employee details table :	empdetail
Issued books details table :	issuedetail

Temporary Data Structure:

Data which can be updated or modified on a regular basis:

Bookid , issueto , issueby , book -status.

4.5 Object- Oriented Design

We have 4 main class :

Employee :	EmpRegister()
	gettingEmpDetails()
	Employee()

Student :	studentRegister()
	gettingStuDetails()
	Student()

Books :	searchBook()
	ViewBooks()
	returnBook()
	addBooks()
	deleteBook()

The issued books :	issueBooks()

Main functions :

def gettingEmpDetails()

: encapsulates all the functions required to save the employee details.

def gettingStuDetails():

: encapsulates all the functions required to save the student details.

def gettingLoginDetails():

: consists of required to save the login details such as user name and password

def EmpRegister():

: gets the details of the employee while registering

def studentRegister():

: gets the details of the student while registering

4.6 Procedural Approach

Mainly focuses on basic operations in a library like adding new books, and updating new information, searching books and members and return books.

Step 1: login as student or employee

Step 2: Enter your details as a student or employee

Step 3: Under Employee :

Add a book

Issue a book

Delete a book

Return a book

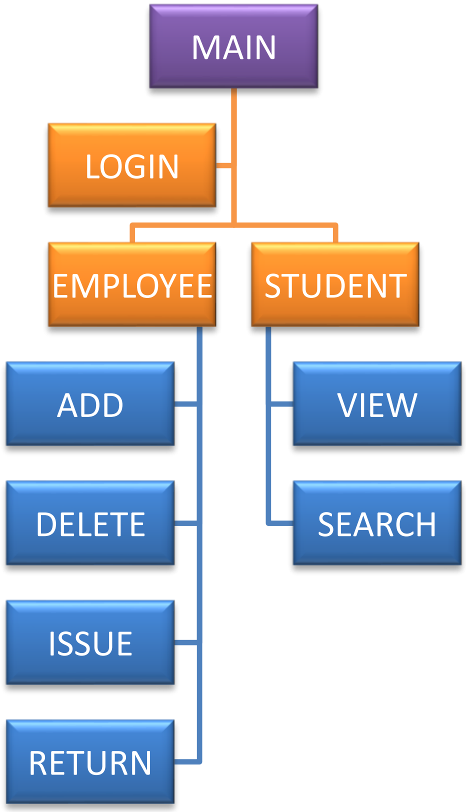
Step 4: Under student :

View a book

Search a book

5 Detailed Design:

5.1 Architechture



5.2 Description and Components

- Main componenet starts wilth a login screen

- Login - here employee and student can login using id given to them

- Student can access view and search component

- Employee can access add/delete/issue/return as well as student components

5.3 Software Interface Design

- Software Interface design is done using python tkinter module.

5.4 Modules

- LMS uses MySQL for database as backend
- LMS uses python-tkinter as frontend

---

6 Interface Design

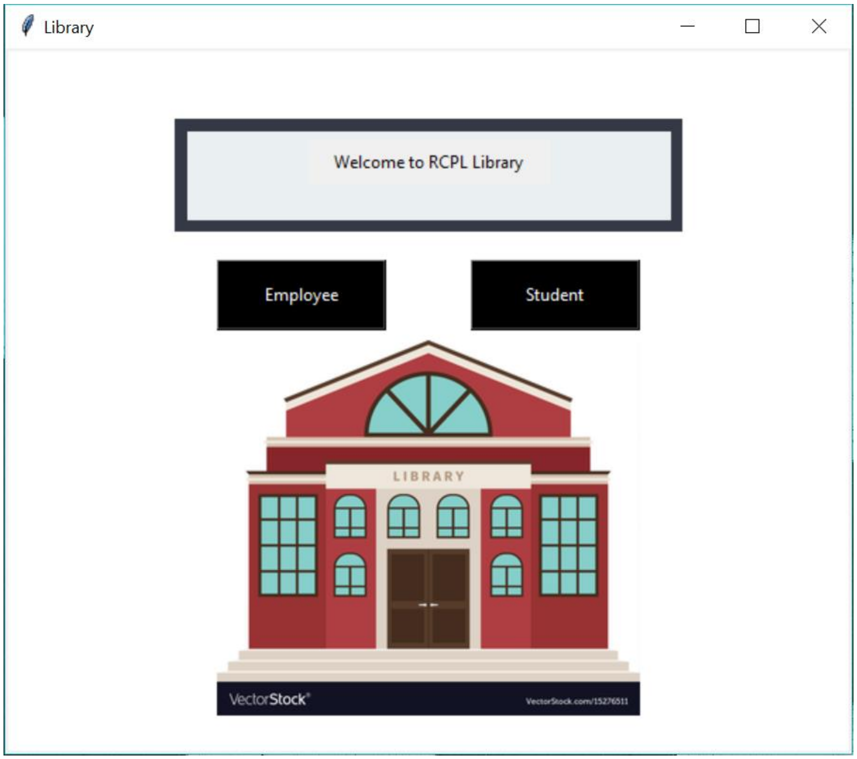
6.1 Interface Description

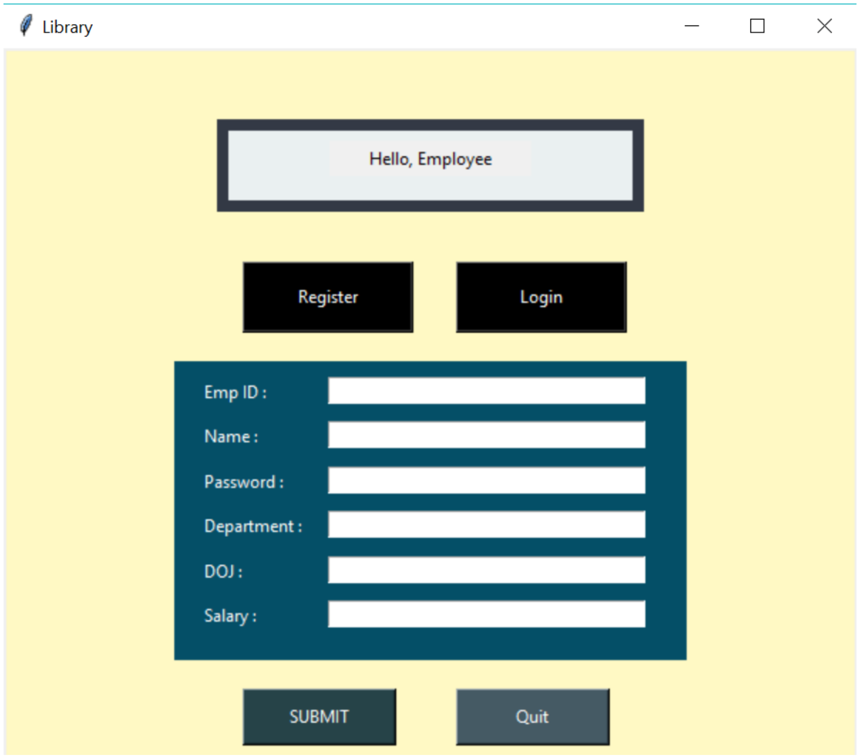
- Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

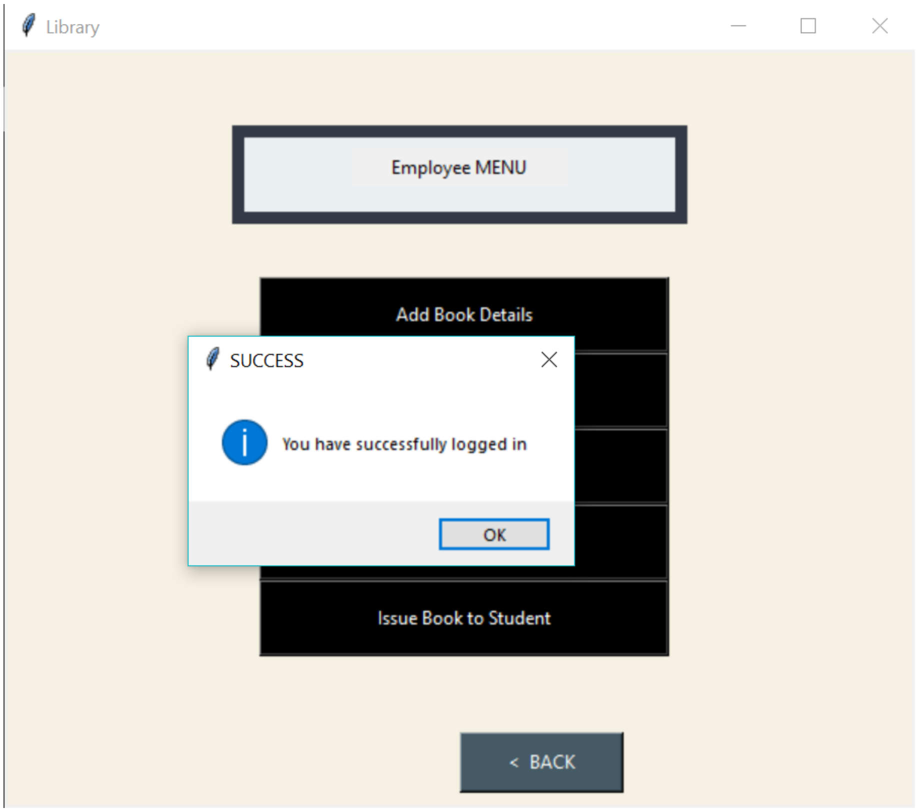
---

7 User Interface Design

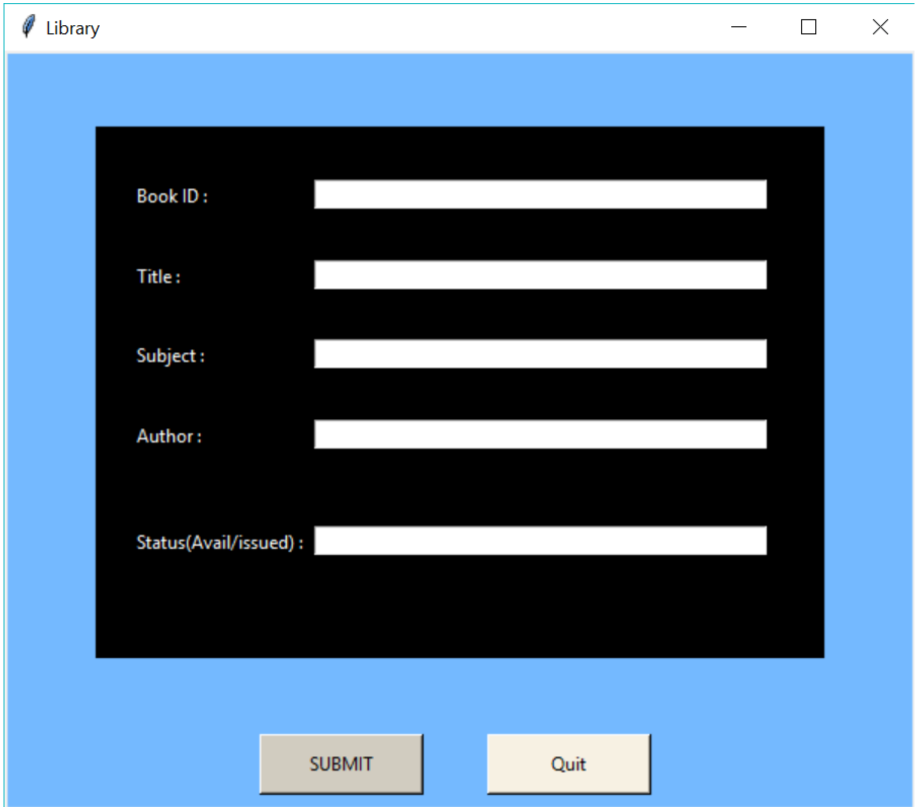
7.1 UI

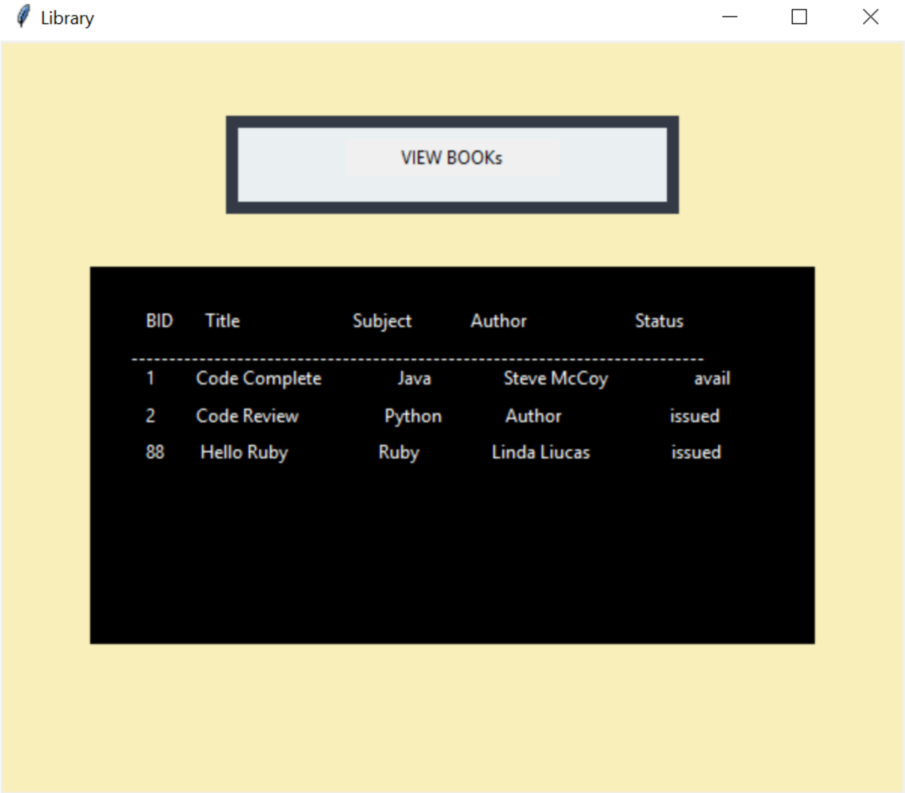

Library
Welcome to RCPL Library
Employee
Student
LIBRARY
VectorStock®
VectorStock.com/15276511


Library
Hello, Employee
Register
Login
Emp ID :
Name :
Password :
Department :
DOJ :
Salary :
SUBMIT
Quit

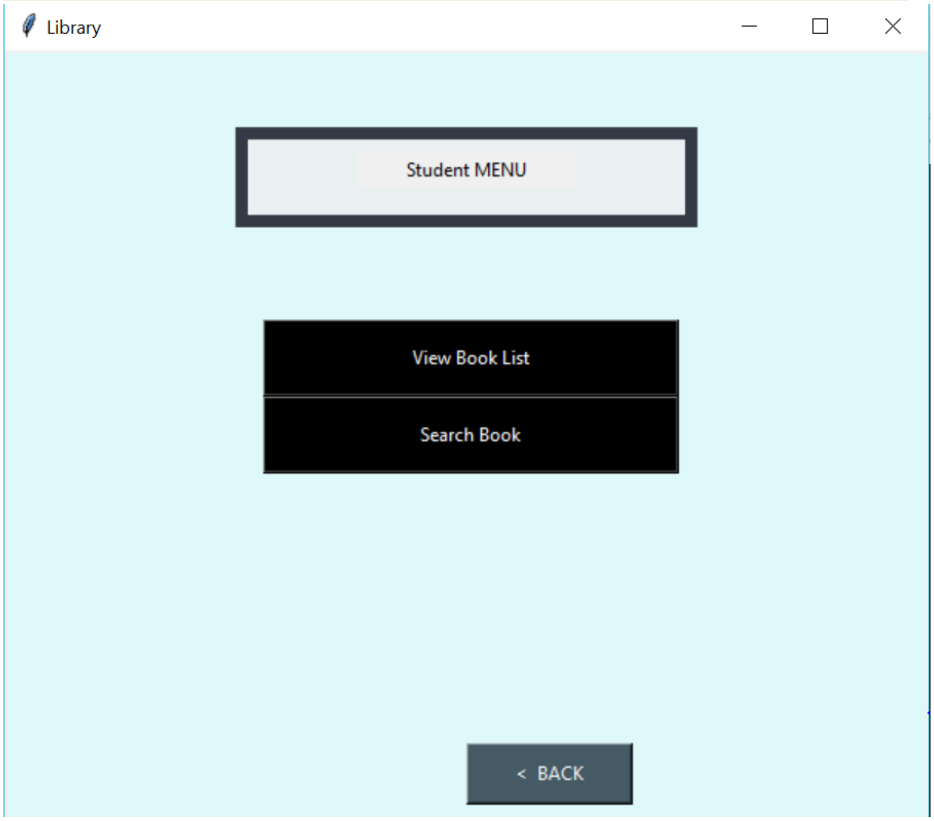


You have successfully logged in





```
BID   Title            Subject     Author          Status
-----------------------------------------------------------------
1     Code Complete      Java        Steve McCoy        avail
2     Code Review        Python      Author           issued
88    Hello Ruby         Ruby        Linda Liucas     issued
```



7.2 Components Available

- Main
- Login
- Student:
    - View and
    - Search component
- Employee:
    - Add
    - Delete
    - Issue
    - Return as well as
    + Student components

7.3 Development description

- UI development is done by using tkinter module in python.
- Tkinter is the standard GUI library for Python