SEPTEMBER 14, 2020

LIBRARY MANAGEMENT SYSTEM

CEP OOP PROJECT

CS-116

HANIYA MAQSOOD CS-19018

HALEEMA AFSHAN KHAN CS-19023

RUMAISA MARYAM CS-19024

SUBMITTED TO : DR. MARIA WAQAS

DEPARTMENT OF COMPUTER & INFORMATION SYSTEMS ENGINEERING
BACHELORS IN COMPUTER SYSTEMS ENGINEERING

Course Code: CS-116
Course Title: Object Oriented Programming
Complex Engineering Problem
FE Batch 2019, Spring Semester 2020
Grading Rubric
TERM PROJECT

Group Members:

Student No.	Name	Roll No.
S1		
S2		
S3		

CRITERIA AND SCALES			Marks Obtained		
			S1	S2	S3
Criterion 1: Does the application meet the desired specifications and produce the desired outputs?					
0 – 1	2 - 3	4 – 5			
The application is producing incorrect outputs.	The program produces correct outputs but does not display them correctly.	The program works and meets all of the specifications.			
Criterion 2: How well is the code organization?					
0 – 1	2 - 3	4 – 5			
The code is poorly organized and very difficult to read.	The code is readable only by someone who knows what it is supposed to be doing.	The code is well organized and very easy to follow			
Criterion 3: How friendly is the application interface?					
0 – 1	2 - 3	4 – 5			
The application interface is difficult to understand and use.	The application interface is easy to understand and but not that comfortable to use.	The application interface is very easy to understand and use.			
Criterion 3: How did the student answer questions relevant to the task?					
0 – 2	3 - 7	8 – 10			
The student answered few questions relevant to the solution.	The student answered most of the questions relevant to the solution.	The student answered all the questions relevant to the solution.			
Criterion 4: Does the report adhere to the given format?					
0 – 1	2 - 3	4 – 5			
The report does not contain the required information. OR The organization of the report is poor.	The report contains the required information only partially. OR The organization of the report follows the given format only partially.	The report contains all the required information and completely adheres to the given format.			
		Total Marks:			

Problem Description

The application that we have developed using object oriented programming in python is named as "**Library Management system or Virtual library**".

Our virtual library is basically going to using GUI for its interface. There are two main categories of people that can use this one is the user and the other is the librarian.

The functionality of our library includes:

1) Create account (user), sign in (users, librarian)

2) Store record of users (books they bought, email)

3) Show records of users (can be seen by manager, user)

- **"Custom functionality"**

4a) Show all the books in the library

4b) Show all the books "available" in the library, displays prices after book has been added

5) Add books

6) Lend books

7) Reserve book

8) Search the book by its name or author

9) Checking stock (manager)

10) Updating stock (manager)

11) Add book to library

12) Remove book from cart the library

13) Reissue book

14) Return book

15) Payment method

Distinguishing features of our project:

Our library management system not only fulfills the minimum requirements of the project but it also goes on to produce some really cool features which enhance our project. Instead of only using four features of oop we have tried to use all the features like we have used composition, inheritance, exceptions, abstract class, method overriding. We have tried to create a payment method that takes users card number as input and prompts him all the dues have been cleared. We have maintained a user log where all the information regarding the user's credentials is stored. Apart from that we have created a user history log that gives us information regarding the activity of the user. We have developed this whole application on GUI using python's Tkinter library. We have explored various features of this library in our project to enhance it and make it more sophisticated. We have introduced a really intriguing function of lending books into our application. This function takes the book name as input after taking the user id and lends the book for a specific time period. Since we are lending a book we have also introduced a "Fine" function that calculates the fine after a certain due date.

Flow of our project including class diagram:

After downloading all the attached .py files along with the text files, the file that should be first run is the A1_CS19018_4.py file. This file contains and binds all the code together.

We have distributed the working of our application into five files. The first file A1_CS19018_1.py contains all the functionality of the user. In this file the first defined class is the Address class, this class is used to print the address of the library. Then we have defined the person class, this class is used to take the name and email of the user as input and then we have initialized a dictionary so that we can place the name and email of the user in it for record. This class also assigns a unique number called id to a user which can be used to avail various features that can be used by a user.

2) Then we have imported a date time module and defined a date class. This date class keeps track of the time the book was issued or lended and the due date i.e. the date it has to be returned on. This class is also used to issue fine to the user on late return of the book.

3) Then we have a Book Reservation class that is used to reserve, add, remove and fetch details about reservation as well. A part from that we have defined a book item class that contains a dictionary having all the books available in the library.

4) We have created a book lending class that is inheriting from book item and date class. This class is used to check if the book that user wants to lend is available in the library or not. And the date class gives the due date and issue date.

ii) The second file is A1_CS19018_2.py , this file is used to contain all the information regarding the librarian.

1) We have first imported the abstract class module and passed it as a parameter to our account class, making it abstract, the reason for this is that it used to set the password, name and email of the librarian so we have made it abstract. Further instance of this class cannot be created.
2) Then we have defined a librarian class that is inheriting from our abstract account class. In this class we have overridden the constructer of the account class. This class is also used to add or update the stock or check the stock.

iii) Then the third file is A1_CS19018_3.py. This file contains all the GUI based classes of our application. This file contains all the GUI based classes related to the functionality provided to the user.

1) We have first imported the A1_CS19018_1.py file that contains all the functionality of the user and then one by one converted every single class into gui .We first imported all the features of the A1_CS19018_1.py into our A1_CS19018_3.py file.

2) This fie basically contains two classes. One is the user class and the other is the main screen class. The user class reads the lines of the file created with the name of user log and gives out a new id to every person who registers. Then we have the main screen classes and this class contains all the functions that put up the gui screens on the interface,

3)Functions like LIBRARY_INFO, Address, Check_Fine, Exit, history, Lend_Book, lend_id, log1, book_reserve , add_books , fetch_reservation_detail , payment , GeneralInfo , displaybook are used to as the name suggests display the information regarding the

library ,address,fine,history of the user,book lending,book reservation, adding books, fetch book, details, payment etc in gui format.

iv) Then we have a file named A1_CS19018_4.py, this file deals with all the login functionality inherited by the application.

1) This file is divided into two classes. One is the user log class that checks if any id is assigned to the user, if so it will increment it and give a new id and if not than it will assign the id 1 to the user.

2) Then we have a class called main class. This class contains all the gui based functions related to gui. This class has the functions that asks the use to register or log in and after that it verifies the entered credentials in case of the user login through the user history file.

V) for the librarian class gui we have defined a file named A1_CS19018_5.py that has a class named lib screen that is used to create the gui of our librarian class. In this class we have imported and used the functionality of the A1_CS19018_2.py.

After defining and constructing all the classes we run the A1_CS19018_4.py . This file calls all the other files in itself by importing them as modules.

In Short:

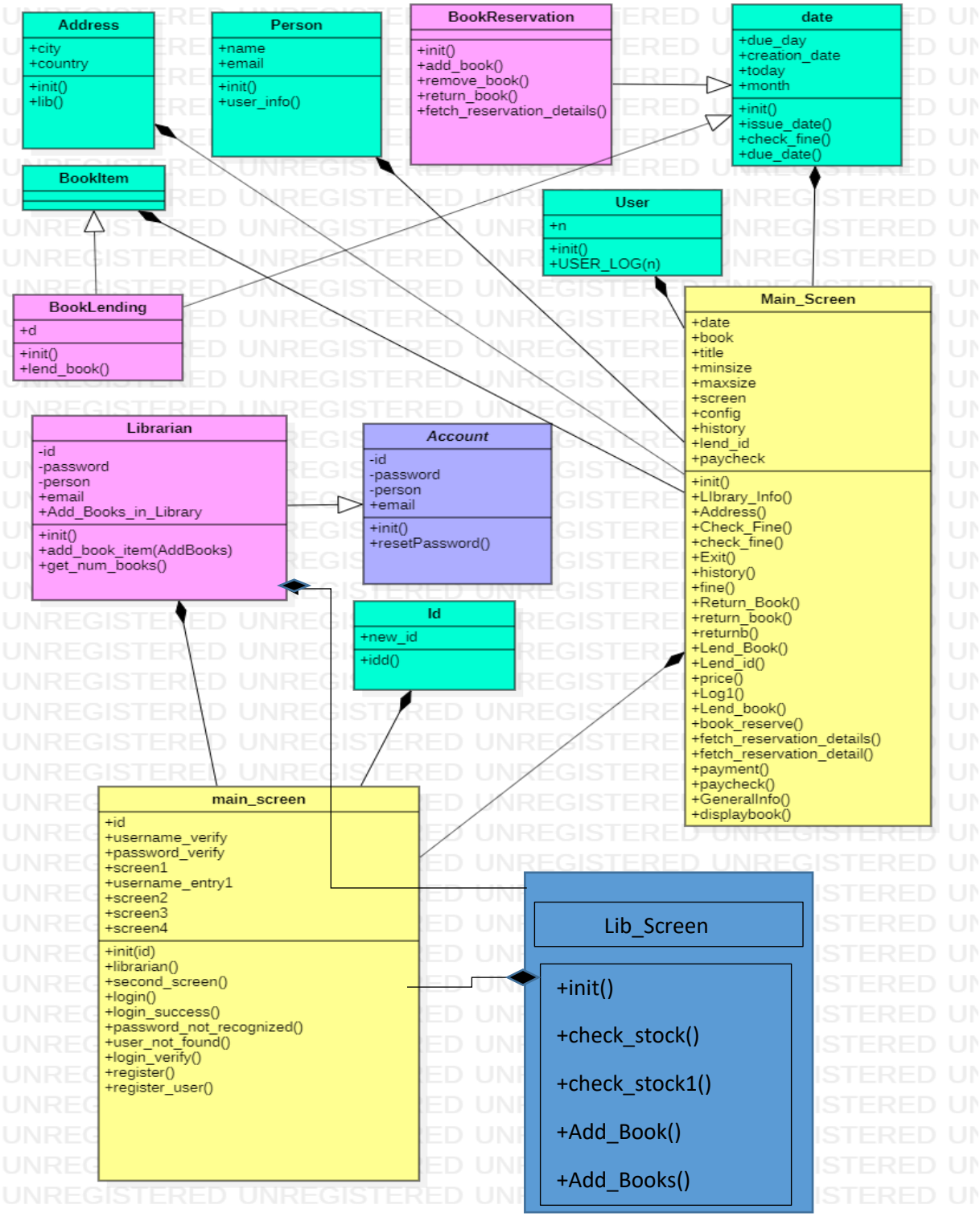
A1_CS19018_1.py: Contains all the functionality of the user

A1_CS19018_2.py : Contains all the functionality of the librarian

A1_CS19018_3.py : Contains the GUI classes of the user

A1_CS19018_4.py : Contains all the GUI classes for login

A1_CS19018_5.py: Contains all the GUI classes of the librarian



CLASS DIAGRAM

Most challenging part for us while working on the project:

The most challenging part no doubt was the conversion of the code in oop to GUI, during this we learnt a lot of the new features that we didn't know existed before .Also the need to put in some extra functionality really upped the game in terms of the usage of the gui. The part that was one of the most challenging would also include the designing part as well, binding all the constraints together and using them in a harmonious way was indeed a challenge in its own.

All in all we really tried our best to put up something that does justice to the time it took to complete and all the things that have been taught to us till this point onwards.

Also we faced quite a bit of challenge while maintaining the log of the user, because maintaining it while retrieving information from it and using that info to assign new ids to the users was really hectic

Any new thing learnt in Python while working on the project:

There were in general a lot of things that we learned but the most prominent among them would be the features of the tkinter library of python. Also converting a code in oop to gui itself was a new thing that we haven't done before.

Apart from that we used a lot of the features of the gui library for the first time like top level screen up, bringing one screen on another one, inserting background image, delete end feature etc.

Future expansions:

A Project can never be deemed as complete or that no more functionality can be added to it. The same is true for our program as well. A few features that can be added to this application to expand it further would be:

1) Sending warnings to the users to return the lended book before due date.

2) Reviews section about the book

3) Reset password option available to user

4) Creating a catalog for the library

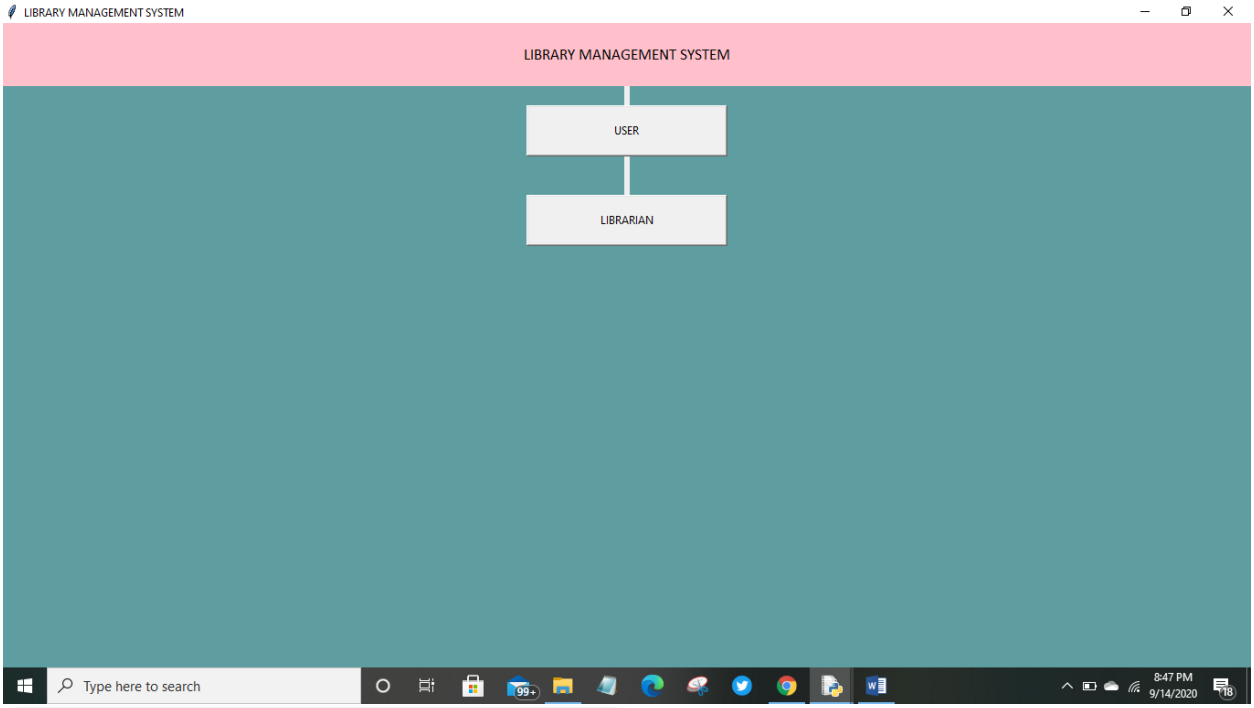
5) Sending notifications to the users regarding new upgrades etc.

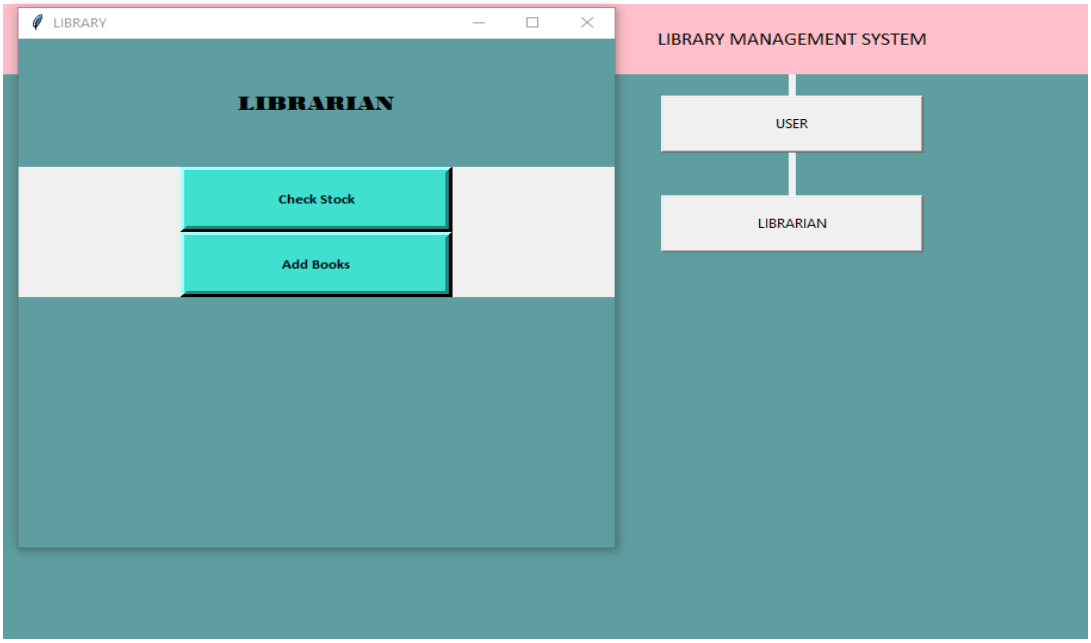
6) Restriction in lending more than 10 books etc.

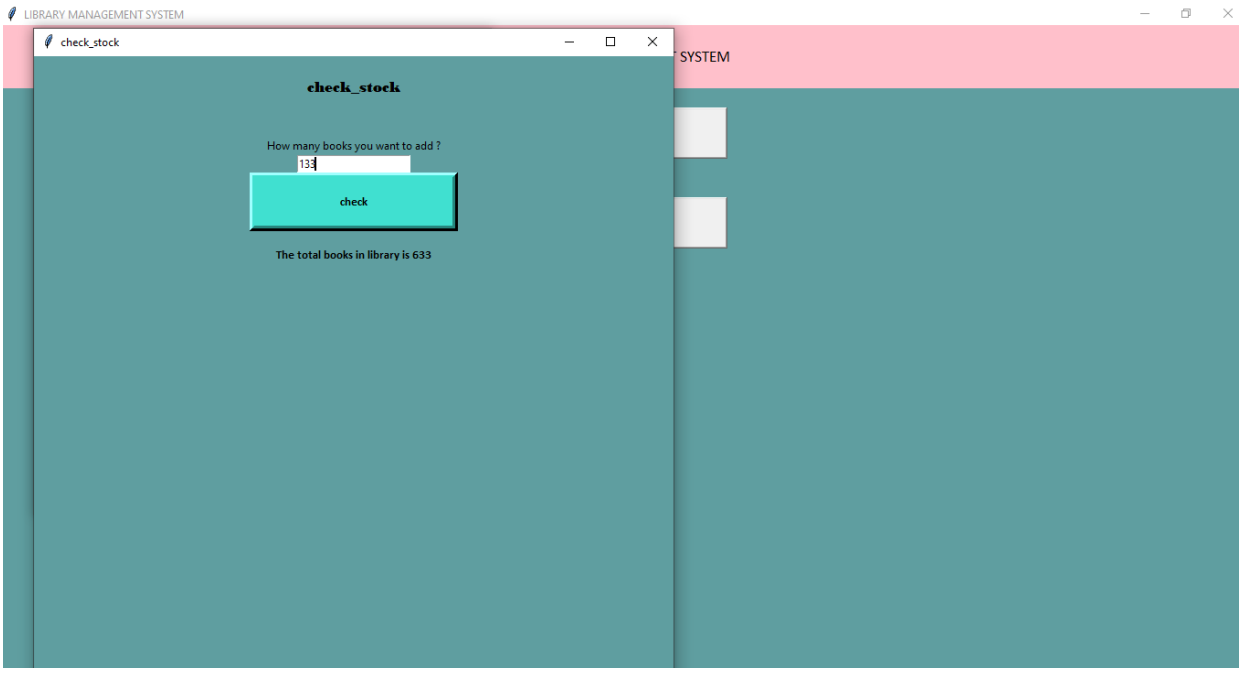
7) Creating a web page for the library

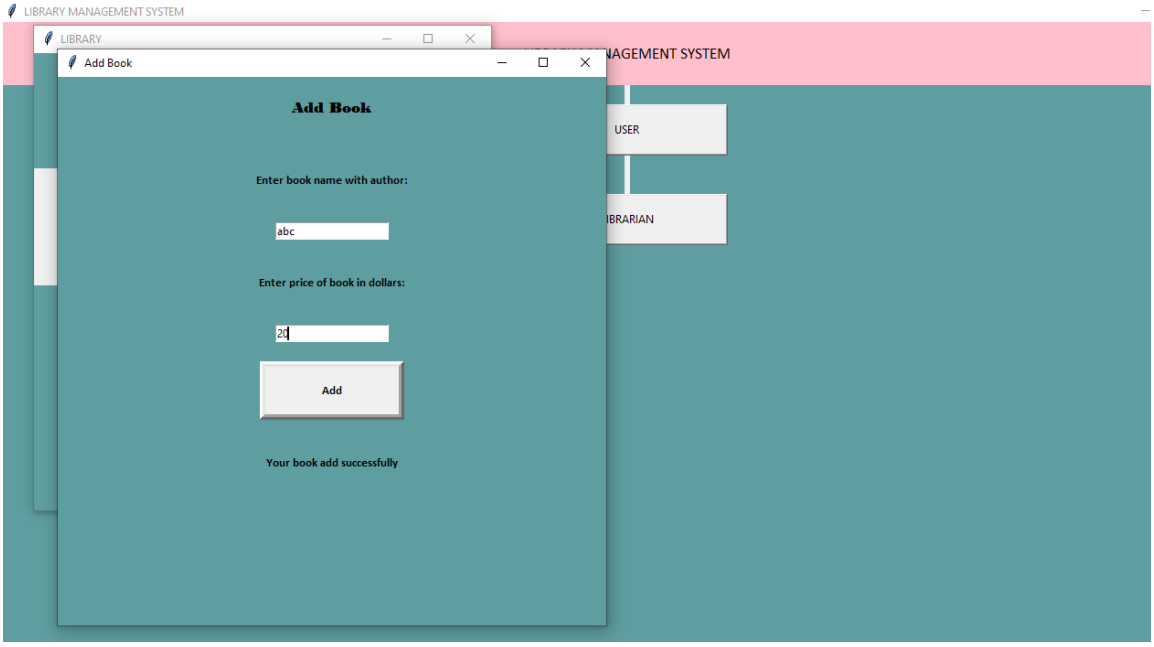
There are a lot of things that can be added but the above mentioned features would enhance the end product more.

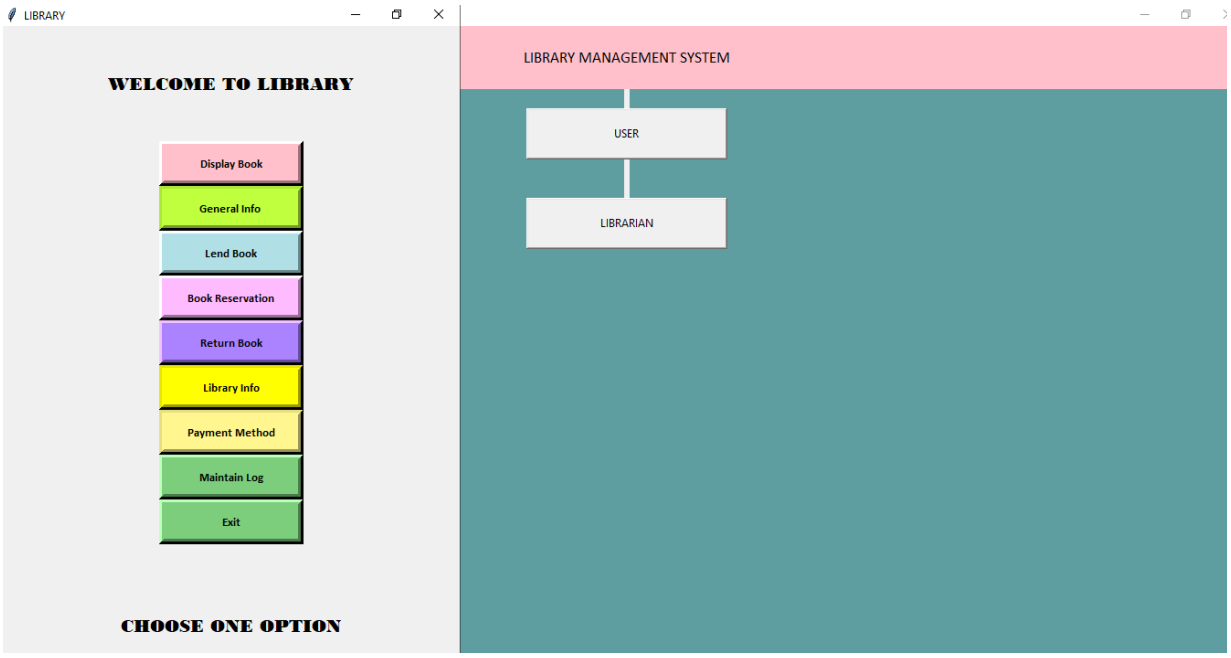
TEST RUNS


LIBRARY MANAGEMENT SYSTEM
LIBRARY MANAGEMENT SYSTEM
USER
LIBRARIAN
Type here to search
8:47 PM
9/14/2020


LIBRARY
LIBRARIAN
Check Stock
Add Books
LIBRARY MANAGEMENT SYSTEM
USER
LIBRARIAN

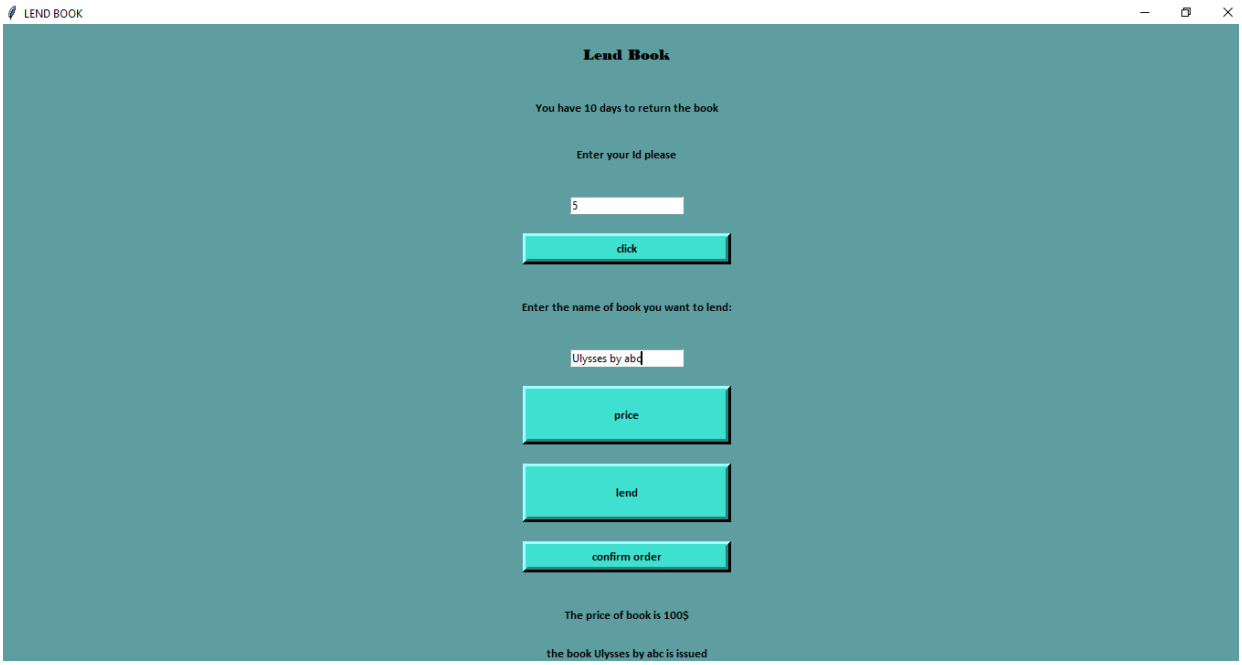

LIBRARY MANAGEMENT SYSTEM
check_stock
check_stock
How many books you want to add ?
133
check
The total books in library is 633
SYSTEM

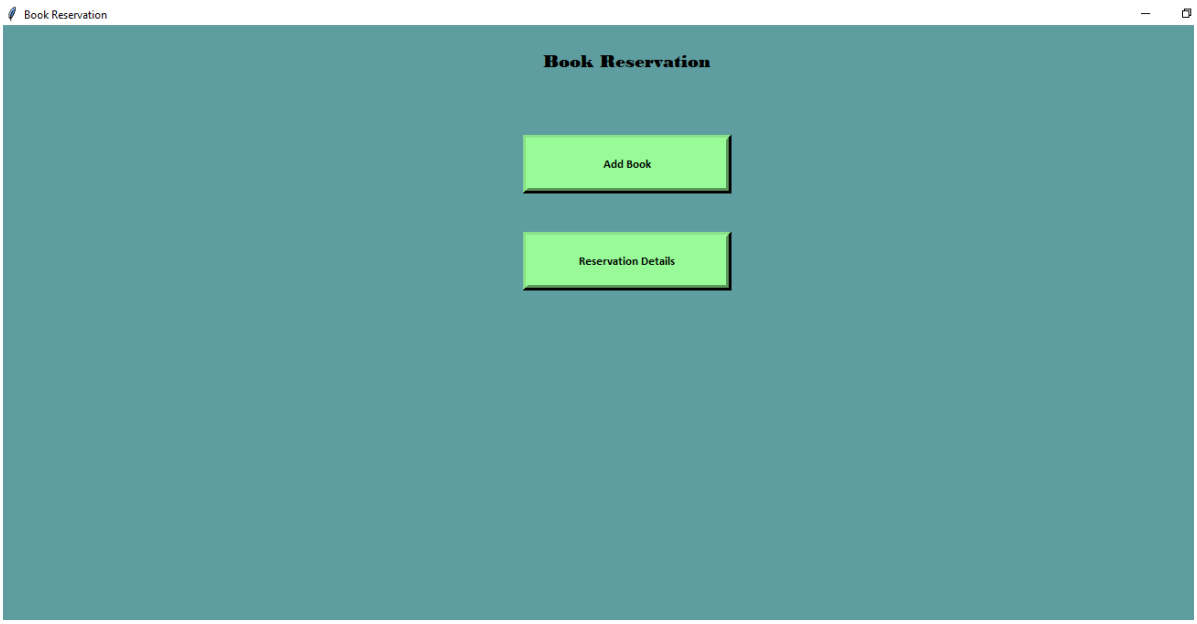

LIBRARY MANAGEMENT SYSTEM
LIBRARY
Add Book
Add Book
Enter book name with author:
abc
Enter price of book in dollars:
20
Add
Your book add successfully
USER
IBRARIAN

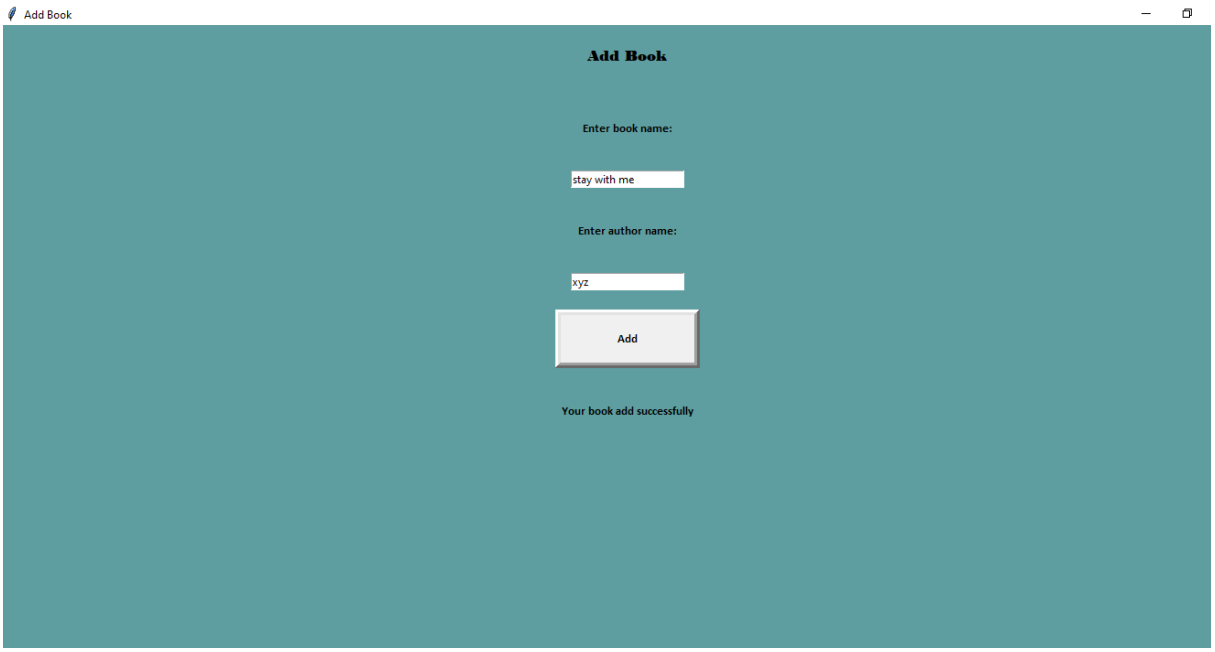

LIBRARY
WELCOME TO LIBRARY
Display Book
General Info
Lend Book
Book Reservation
Return Book
Library Info
Payment Method
Maintain Log
Exit
CHOOSE ONE OPTION
LIBRARY MANAGEMENT SYSTEM
USER
LIBRARIAN

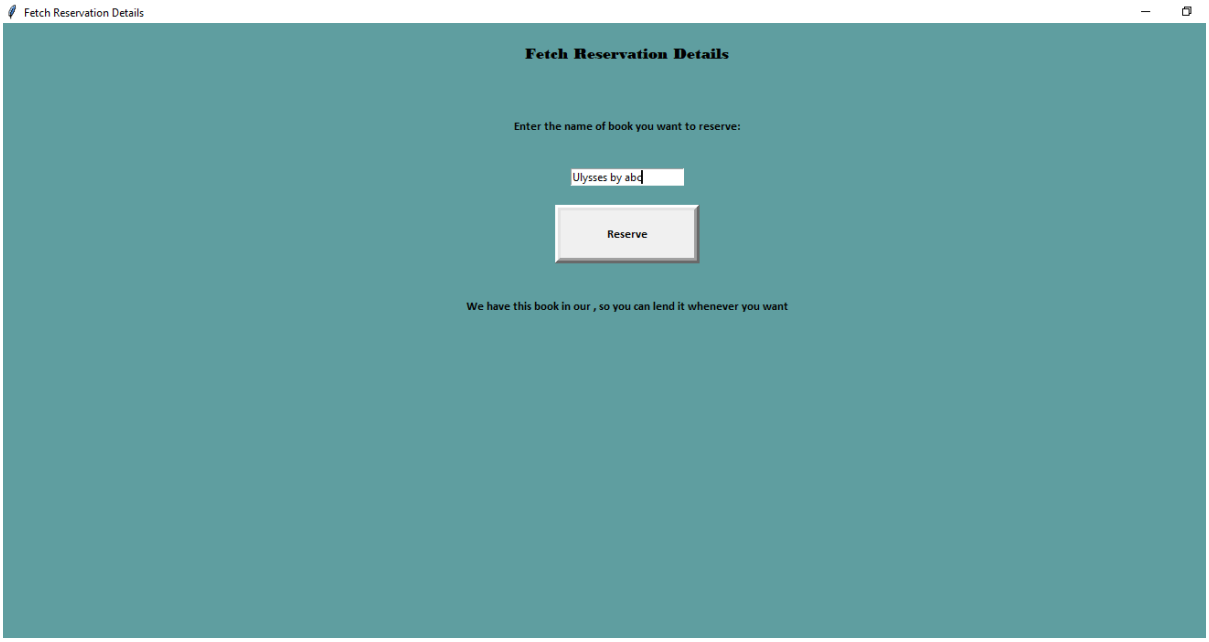

Display Book
Display Book
silent patient by Alex Michaelides
alchemist by polu coello
One Hundred Years of Solitude by Gabriel García Márquez
War and Peace by Leo Tolstoy
Lolita by Vladimir Nabokov
In Search of Lost Time by Marcel Proust
Ulysses by abc
The Great Gatsby by
One Hundred Years of Solitude by Gabriel Garcia Marquez
Hamlet by William Shakespeare
harry potter by JK Rowling
The Handmaid's Tale by Margaret Atwood

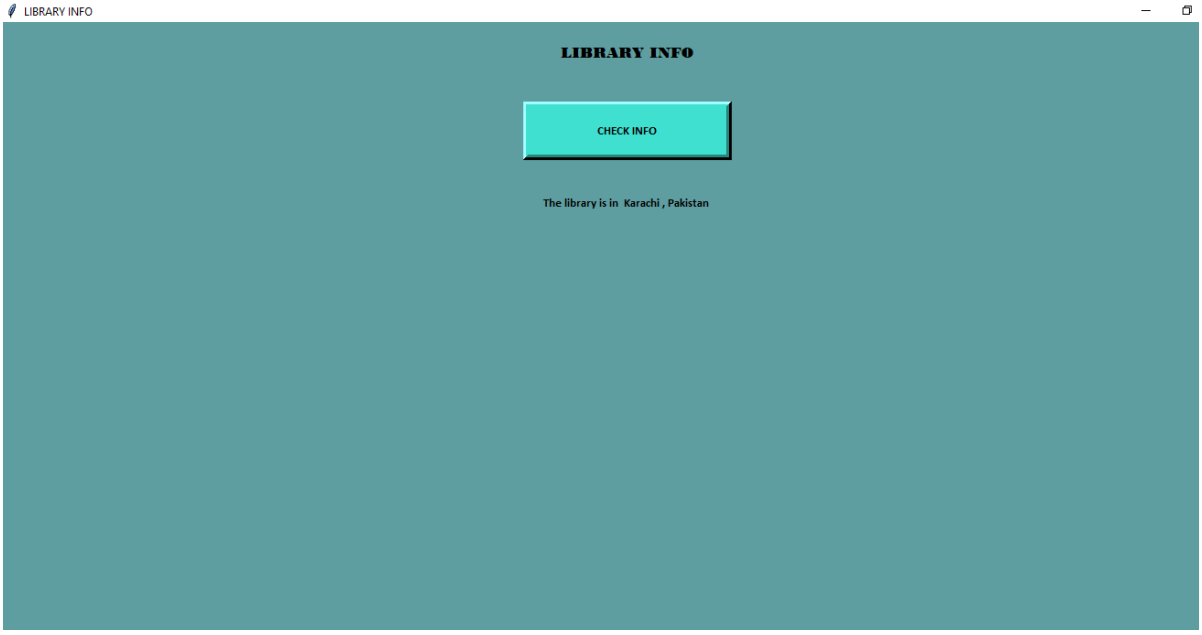

General Info
General Info
Welcome to Student Library
You can lend books for 10 days
After 10 days fine will be applicable to all students
You can reserve book and Add books of your choice
There is a payment method by Account

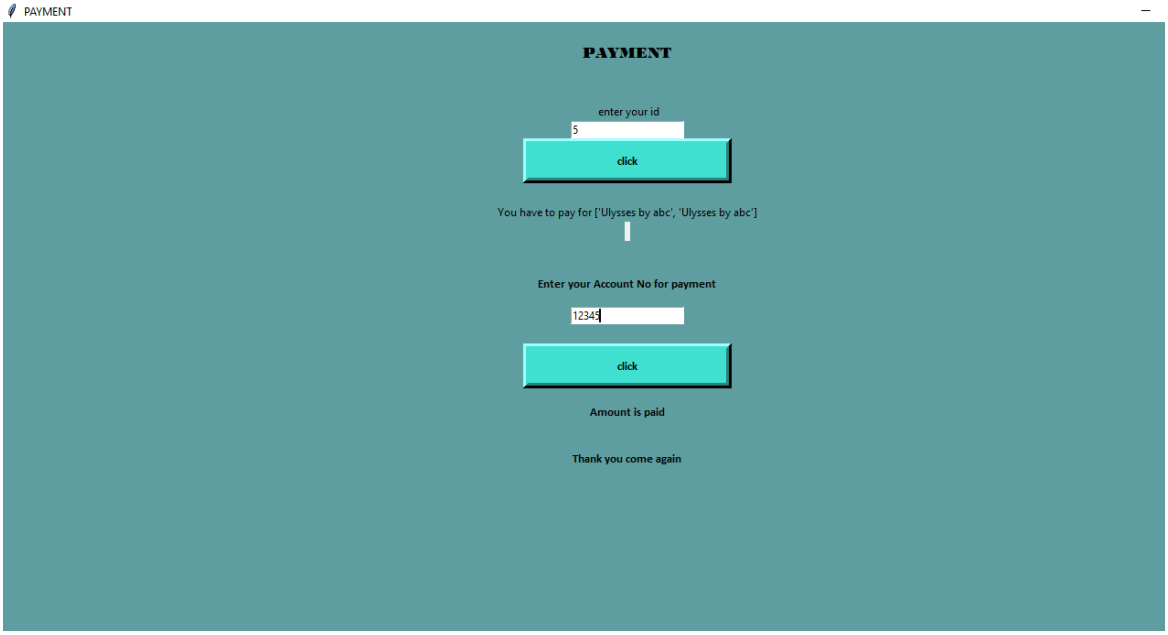

LEND BOOK
Lend Book
You have 10 days to return the book
Enter your Id please
5
click
Enter the name of book you want to lend:
Ulysses by abc
price
lend
confirm order
The price of book is 100$
the book Ulysses by abc is issued


Book Reservation
Book Reservation
Add Book
Reservation Details

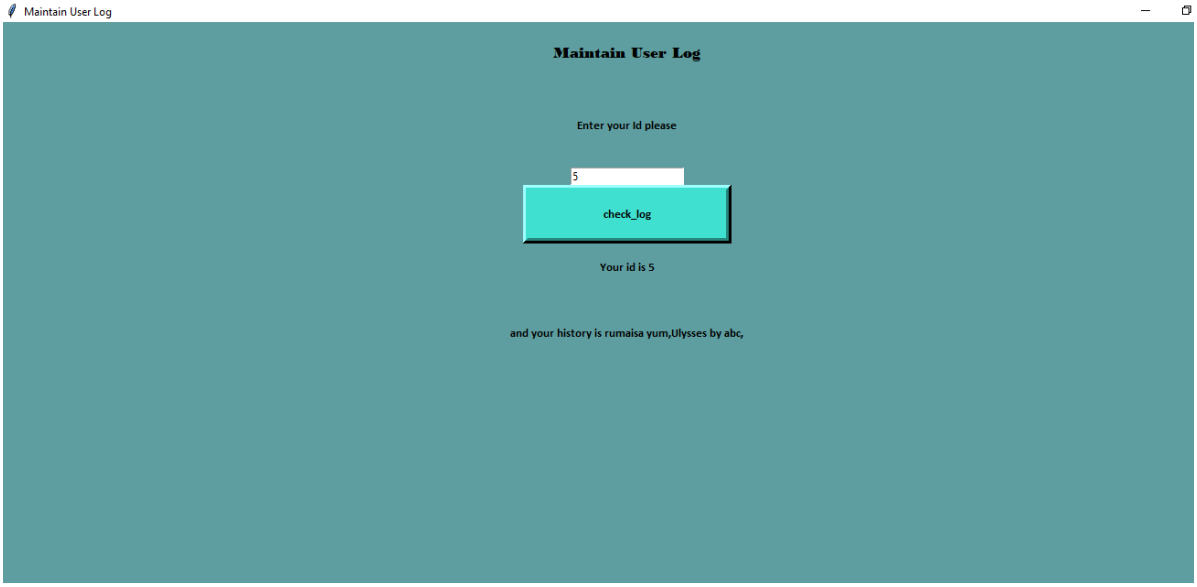

Add Book
Add Book
Enter book name:
stay with me
Enter author name:
xyz
Add
Your book add successfully


Fetch Reservation Details
Fetch Reservation Details
Enter the name of book you want to reserve:
Ulysses by abc
Reserve
We have this book in our , so you can lend it whenever you want


LIBRARY INFO
LIBRARY INFO
CHECK INFO
The library is in Karachi , Pakistan


PAYMENT
PAYMENT
enter your id
5
click
You have to pay for ['Ulysses by abc', 'Ulysses by abc']
Enter your Account No for payment
12345
click
Amount is paid
Thank you come again


Return Book
Return Book
return book
enter your Id please:
5
click
you lend the book ['Ulysses by abc', 'Ulysses by abc']
Enter book name you want to return
Ulysses by abc
check fine
Welcome !!! Now you can return the book
Enter today date in local format dd/mm/yy
19/09/20
click


Maintain User Log
Maintain User Log
Enter your Id please
5
check_log
Your id is 5
and your history is rumaisa yum,Ulysses by abc,