LIBRARY MANAGEMENT SYSTEM

TABLE CONTENTS

INTRODUCTION :

Library management system is a project which aims in developing a computerized system to maintain all the daily work of library.

AIM :

“Online management administrative work in library”.

OBJECTIVE :

The main objective of the Library Management System is to manage the details of books, students, book issue and book return.

NEED OF THE SYSTEM :

There is always a need of a system that will perform online management of books, issue books, and submit.

This system will reduce the manual operation required to maintain all the records and also generates the various reports for analysis.

HARDWARE AND SOFTWARE REQUIREMENTS

CONTENTS :

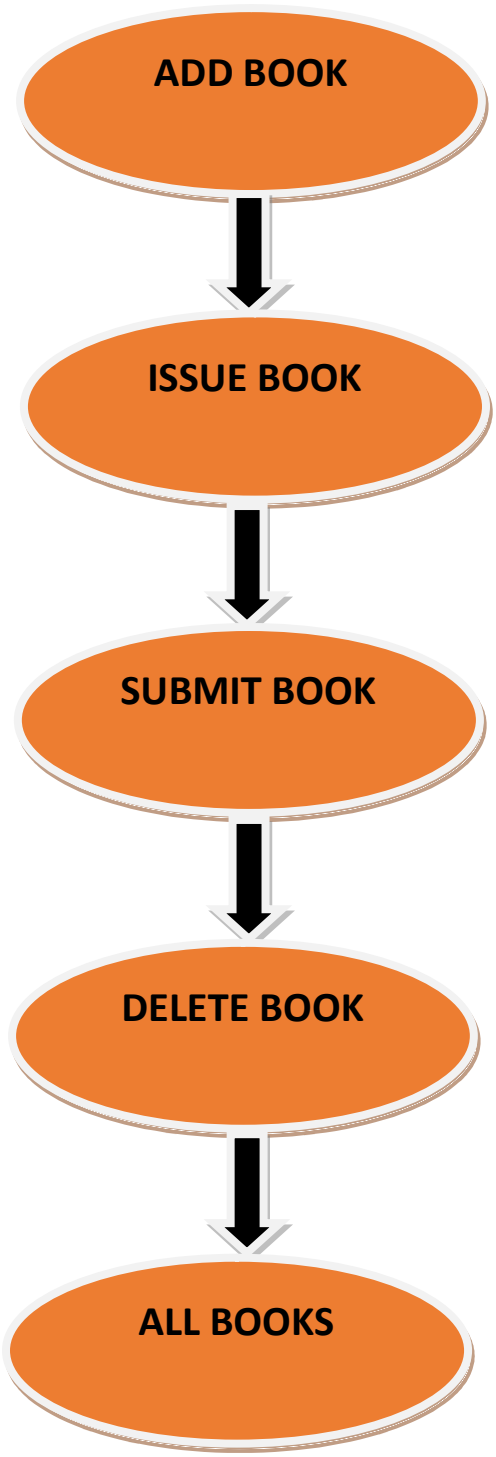
❖ Software Requirement :

- Microsoft Windows 7/8/9/10.
- Xampp server (MySQL, Apache, PHP)
- Notepad or any other text editor.
- Chrome or any other browser.

❖ Hardware Requirement :

- Intel Processor 2.0GHz or above.
- 2 GB RAM or more.
- 160 GB or more Hard Disk Drive or above.

CASE DIAGRAM :



ACTUAL CODE :

```
# MYSQL connectivity TO SERVER
import mysql.connector
mydb = mysql.connector.connect(host = "localhost",
                               user = "root",
                               password = "",
                               database = "lmanage"
                               )

#This is a functions

def addbook():
    bname = input("Enter book name:")
    bcode = input("Enter book code:")
    total = input("Total books:")
    sub = input("Enter subject:")
    data = (bname,bcode,total,sub)
    sql = "Insert into books values(%s,%s,%s,%s)"
    c = mydb.cursor()
    c.execute(sql,data)
    mydb.commit()
    print(".......")
    print("Data Entered Successfully")
    main()

def issueb():
    name = input("Enter name:")
    regno = input("Enter Regno:")
    code = input("Enter book code:")
    date = input("Enter date:")
    sql =  "Insert into issue values(%s,%s,%s,%s)"
    data = (name,regno,code,date)
    c = mydb.cursor()
    c.execute(sql, data)
    mydb.commit()
    print(".......")
    print("Book issued to:", name)
    main()
    bookup(code, -1)

def submitb():
    name = input("Enter name:")
    regno = input("Enter Regno:")
    code = input("Enter book code:")
    date = input("Enter date:")
    sql = "Insert into submit values(%s,%s,%s,%s)"
    data = (name, regno, code, date)
    c = mydb.cursor()
    c.execute(sql, data)
    mydb.commit()
    print(".......")
```

```
    print("Book submitted from:",name)
    main()
    bookup(code,1)

def bookup(code,u):
    sql = "select TOTAL from books where BCODE= %s"
    data = (code,)
    c = mydb.cursor()
    c.execute(sql,data)
    myresult = c.fetchone()
    t = myresult[0] + u
    sql = "update books set TOTAL = %s where BCODE = %s"
    d = (t,code)
    c.execute(sql,d)
    mydb.commit()
    main()

def dbook():
    ac = input("Enter book code")
    sql = "delete from books where BCODE = %s"
    data = (ac,)
    c = mydb.cursor()
    c.execute(sql, data)
    mydb.commit()
    main()

def allbook():
    sql = "select * from books"
    c = mydb.cursor()
    c.execute(sql)
    myresult = c.fetchall()
    for i in myresult:
        print("Book name:",i[0])
        print("Book code:", i[1])
        print("Total:", i[2])
        print(".........")
        main()

def main():
    print("""........LIBRARY MANAGEMENT.........
    1. Add book
    2. Issue book
    3. Submit book
    4. Delete book
    5. Display All book
    """)
    choice = input('Enter task no:')
    print("..............")
    if (choice == "1"):
        addbook()
    elif(choice == "2"):
        issueb()
    elif (choice == "3"):
        submitb()
    elif (choice == "4"):
        dbook()
    elif (choice == "5"):
```

```
        allbook()
    else:
        print("Wrong choice")
        main()

# To create a username and generate randon password.
def password():
    import random
    ps = random.randint(00000,100000)

    user = input("Enter username:")
    print("Your password is:",ps)

    verify = input("Enter password:")

    if verify == str(ps):
        main()
    else:
        verify != str(ps)
        print("Wrong password")
        password()

password()
```