**"Library Management System Using Python"**

A

Project Report

Submitted to

**Dr. Babasaheb Ambedkar Technological University, Lonere.**

For the Degree of

**Bachelor of Technology (B. Tech.)**

In

**Electronics & Telecommunication Engineering**

Under the Faculty of Science and Technology

Submitted By

**Mr. Somnath Dattu Trimal**

**PRN.: T2163211372508**

Under the Guidance of

**Prof. Wagdarikar A.U.**

**Department of** **Electronics & Telecommunication Engineering**



NAAC Accredited and ISO 9001:2015 Certified

**Vidya Vikas Pratishthan**

**Institute of Engineering & Technology, Solapur.**

Year: 2022-23

NAAC Accredited and ISO 9001:2015 Certified

**Vidya Vikas Pratishthan**

**Institute of Engineering & Technology, Solapur.**

**(***Affiliated to Dr. Babasaheb Ambedkar Technological University, Lonere)*



**CERTIFICATE**

*This is to certify that,* ***Mr./Ms.* Somnath Dattu Trimal*****(PRN:* T2163211372508*)****, a bonafide student of B. Tech. Final Year (VIII Semester), Department of* Electronics & Telecommunication*Engineering,* ***Vidya Vikas Pratishthan Institute of Engineering & Technology, Solapur****, has successfully completed project work titled* **"Library Management System Using Python"** *in the partial fulfillment of degree course* ***Bachelor of Technology (B. Tech.)*** *in* **Electronics & Telecommunication *Engineering*** *of* ***Dr. Babasaheb Ambedkar Technological University, Lonere*** *during the* ***academic year 2022-2023****.*

**Prof. WagdarikarA.U.** **Dr. U. S. Mugale**

**Guide and HOD** Principal

**Place:-**

**Date:-**

**External Examiner:-**

**Date:-**

**ACKNOWLEDGEMENT**

I would like to express my sincere thanks to my project guide Prof. Wagdarikar.A.U.**,** Department of Electronics & Telecommunication Engineering, Vidya Vikas Pratishthan Institute of Engineering & Technology, Solapur for his/her systematic & stimulating suggestions and back-up helped me throughout the project work and writing of this project report. It would not have been possible for me to complete this report within a short time without his/he encouragement, guidance and in time assessment.

I extend my sincere thanks to Principal Dr. U. S. Mugale and Prof. Wagdarikar.A.U., Head, Department of Electronics & Telecommunication Engineering, Vidya Vikas Pratishthan Institute of Engineering & Technology, Solapur for his kindly guidance in my project work completion. I am thankful to my family for their constant support and inspiration. At the end I would like to thank all persons who directly or indirectly helped me to complete this project work.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NUMBER** | **CONTENT** | **PAGE NUMBER** |
| 01 | ACKNOWLEDGEMENT | 03 |
| 02 | ABSTRACT | 05 |
| 03 | TABLE OF CONTENTS | 04 |
| 04 | INTRODUCTION | 06 |
| 05 | SOURCE CODE | 07 |
| 06 | OUTPUT | 11 |
| 06 | CONCLUSION | 12 |

**ABSTRACT**

The Library Management System is a software application designed to efficiently manage the operations of a library. This system is developed using Python programming language to provide an intuitive and user-friendly interface for librarians and library staff to automate various tasks involved in library management.

The primary goal of the Library Management System is to streamline the processes associated with library functions such as borrowing, returning, donating and tracking. It aims to simplify and automate routine tasks, thereby saving time and effort for the library staff, as well as enhancing the overall user experience for library patrons.

Moreover, the Library Management System includes a robust search functionality, allowing users to quickly find books based on title. It also supports tracking to find out which book we have borrowed.

The implementation of the Library Management System in Python ensures a portable and platform-independent solution that can be easily deployed across different operating systems.

Overall, the Library Management System in Python aims to enhance the efficiency and effectiveness of library operations, making it an indispensable tool for libraries of all sizes. By automating key processes and providing a user-friendly interface, it enables librarians to focus more on providing quality services to library patrons while efficiently managing library resources.

**INTRODUCTION**

A library management system is an essential tool for efficiently managing the operations and resources of a library. It helps librarians automate various tasks, such as book borrowing, returning, donating and keeping track of borrowed books In this Python project, we will develop a Library Management System to streamline the library's day-to-day operations and enhance its overall efficiency.

By automating these processes, the system will save time, reduce manual errors, and provide accurate and up-to-date information about the library's resources.

Python, a powerful and versatile programming language, is an ideal choice for developing a Library Management System. Its simplicity, readability, and vast array of libraries make it easier to build and maintain software applications. We will leverage Python's object-oriented programming (OOP) capabilities to create modular and reusable code components.

1. Book Management: Librarians can add new books to the system, update existing book information, search for books based on various criteria, and maintain an organized catalog of the library's collection.
2. Member Management: The system will enable librarians to manage member details, including registration, membership renewal, and tracking borrowing history.
3. Circulation: Librarians can handle book circulation efficiently by issuing books to members, tracking record of books name.

**SOURCE CODE**

**INPUT:**

class Library:

def \_\_init\_\_(self, listofBooks):

self.books = listofBooks

def displayAvailableBooks(self):

print(f"\n{len(self.books)} AVAILABLE BOOKS ARE: ")

for book in self.books:

print(" ♦-- " + book)

print("\n")

def borrowBook(self, name, bookname):

if bookname not in self.books:

print(

f"{bookname} BOOK IS NOT AVAILABLE EITHER TAKEN BY SOMEONE ELSE, WAIT UNTIL HE RETURNED.\n")

else:

track.append({name: bookname})

print("BOOK ISSUED : THANK YOU KEEP IT WITH CARE AND RETURN ON TIME.\n")

self.books.remove(bookname)

def returnBook(self, bookname):

print("BOOK RETURNED : THANK YOU! \n")

self.books.append(bookname)

def donateBook(self, bookname):

print("BOOK DONATED : THANK YOU VERY MUCH, HAVE A GREAT DAY AHEAD.\n")

self.books.append(bookname)

class Student():

def requestBook(self):

print("So, you want to borrow book!")

self.book = input("Enter name of the book you want to borrow: ")

return self.book

def returnBook(self):

print("So, you want to return book!")

name = input("Enter your name: ")

self.book = input("Enter name of the book you want to return: ")

if {name: self.book} in track:

track.remove({name: self.book})

return self.book

def donateBook(self):

print("Okay! you want to doante book!")

self.book = input("Enter name of the book you want to donate: ")

return self.book

if \_\_name\_\_ == "\_\_main\_\_":

Delhilibrary = Library(

["Digital Communication", "Microprocessor", "Deep Learning", "Soft Skills", "Python", "Antenna"])

student = Student()

track = []

print("\t\t\t\t\t\t\t♦♦♦♦♦♦♦ WELCOME TO THE VVPIET LIBRARY ♦♦♦♦♦♦♦\n")

print("""CHOOSE WHAT YOU WANT TO DO:-\n1. Listing all books\n2. Borrow books\n3. Return books\n4. Donate books\n5. Track books\n6. exit the library\n""")

while (True):

# print(track)

try:

usr\_response = int(input("Enter your choice: "))

if usr\_response == 1: # listing

Delhilibrary.displayAvailableBooks()

elif usr\_response == 2: # borrow

Delhilibrary.borrowBook(

input("Enter your name: "), student.requestBook())

elif usr\_response == 3: # return

Delhilibrary.returnBook(student.returnBook())

elif usr\_response == 4: # donate

Delhilibrary.donateBook(student.donateBook())

elif usr\_response == 5: # track

for i in track:

for key, value in i.items():

holder = key

book = value

print(f"{book} book is taken/issued by {holder}.")

print("\n")

if len(track) == 0:

print("NO BOOKS ARE ISSUED!. \n")

elif usr\_response == 6: #exit

print("THANK YOU ! \n")

exit()

else:

print("INVAILD INPUT! \n")

except Exception as e: #catch errors

print(f"{e}---> INVALID INPUT! \n")

**OUTPUT**

Enter your choice: 2

Enter your name: Somnath

So, you want to borrow book!

Enter name of the book you want to borrow: Antenna

BOOK ISSUED : THANK YOU KEEP IT WITH CARE AND RETURN ON TIME.

Enter your choice: 5

Antenna book is taken/issued by Somnath.

**CONCLUSION**

In conclusion, the library management system implemented in Python is a powerful and efficient tool for organizing and managing library resources. It offers a range of features and functionalities that streamline the entire process of library operations, from cataloging books to managing borrower information and facilitating seamless checkouts and returns.

The system incorporates key elements such as a user-friendly interface, robust database management, and automated processes, which significantly enhance the overall efficiency and productivity of library staff. By automating routine tasks and reducing manual efforts, the system frees up valuable time and resources, allowing library staff to focus on more critical activities, such as improving user services and expanding the library's collection.

Furthermore, the Python programming language provides a solid foundation for developing the library management system, offering a wide array of libraries and frameworks that facilitate rapid development, code reusability, and easy maintenance. Python's simplicity and readability contribute to a more manageable codebase, making it easier to understand and modify the system as per evolving requirements.

The library management system's benefits extend beyond the library staff, as it provides a seamless and convenient experience for library users. With features such as online catalog search, self-checkout options, and personalized accounts, patrons can effortlessly find and borrow books, renew items, and receive notifications, enhancing their overall library experience.

However, it's important to note that the library management system should be continuously updated and improved to keep up with changing technologies and user expectations. Regular evaluations and feedback from both library staff and users can help identify areas for enhancement and ensure that the system remains efficient, user-friendly, and adaptable.

Overall, the library management system implemented in Python offers a comprehensive solution for effective library administration, improving efficiency, user satisfaction, and facilitating the smooth operation of libraries in the digital age.