**ACKNOWLEDGEMENT**

I would like to express my special thanks of gratitude to my teachers **Mr. Sujit Chatterjee and Mr. Maitraditya Chatterjee**, who gave me the golden opportunity to do this wonderful project of Computer Science on *Library Management System*.

Their help and guidance allowed me to complete the project within the given time frame without facing much difficulty. I have come to learn so many new things while doing this project and I am really thankful to my teachers.

I am also indebted to my team members, my family and my friends for their invaluable support and advice which helped me a lot in finalizing this project within the given time frame.

ARINJOY PRAMANIK

**Introduction of the Project Library Management System:**

The "Library Management System" has been developed to overcome the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardships faced by this existing system. Moreover, this system is designed for the particular need of the institution to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus, by this all it proves it is user-friendly. Library Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources.

Every organization, whether big or small, has challenges to overcome and managing the information of Books, Student, Librarian, Address, Member. Every Library Management System has different Student needs. This system is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. These systems will ultimately allow you to better manage resources.

**Objective of Project on Library Management System:**

The main objective of the Project on Library Management System is to manage the details of Student, Books, Issues, Librarian, Member. The purpose of the project is to build an application program to reduce the manual work for managing the Student, Books, Issues. It tracks all the details about the Issues, Librarian, Member.

**Modules used in the project:**

**Built-in modules:**

* random module
* date and timedelta module from datetime class
* csv module(for reading and writing onto csv files)
* pickle module(for reading and writing onto binary files) **User-created modules:**
* book(where book module(book.py) contains all the functions used in the project)

**Files created in the project:**

Files in this project has been created with the help of a separate file file\_initialise.py. The files are as follows:

* users.dat: This binary file stores username, password of both the users and the admin.
* book\_details.csv: This csv file stores data of all existing books in the library in form of Sl no., book\_name, author, availability.
* issued.csv: This csv file stores data of all books already issued to specific users in form of Sl no, book\_name, issued\_by, date of issuing, date of return.

**Functionalities and User manual:**

* *add\_user():* This function helps to add the user in the library management database by accepting username and password from the user and stores the user’s data in users.dat binary file.

* *login():* This function helps already existing user to login into the library database by matching their corresponding username and passwords and displays a Welcome message if both matches successfully.

* *add\_books():*This function helps the admin in adding the record of each book in the csv file book\_details.csv

* *unique\_book\_no():*This function ensures that the sl no. of each contained in csv file named book\_details.csv is unique by efficient use of random function and set() constructor.

* *issue\_books():*This functions serves the purpose of issuing books as per requested by a specific user. It firsts takes input from user for the sl no. and the name to be issued and using csv.reader() and csv.writer(), makes the availability of the book in book\_details.csv to ‘no’ i.e. unavailable and on other hand adds the book-name,sl no,user who has issued it to another file issued.csv.

Another important aspect of this function is that it stores(in issued.csv) the date when the book is issued and date when it is to be returned by efficient use of date and timedelta module of datetime class.

* *delete\_books():* This function helps the admin to delete already existing books in the library management database as per his requirement by removing those specific books from book\_details.csv by efficiently using csv module.

* *viewbook():*This bidimensional function allows both the user and the admin to view already issued books or all existing books in the library database according to the choice entered.

**Library Management System – Flow of Control**

The Library Management System (LMS) is a python based management system, that provides the functionalities to Add User ( i.e. User Registration or Sign Up), Log In to the LMS Application and the ability to exit (Log Off) from application. The application has two primary program units such as the main program to invoke the application and another python program (books) consisting of various functions definitions to perform Library Management specific tasks. The ‘book’ program is imported into the main application. The description and flow-of-control of the LMS is depicted below:

Invoking the application – Upon involving the application, the use gets application user is asked to choose one of the following options:

**1: Add User**

**2: Log In**

**3: Exit**

**Add User:** If user enterstheoption as ‘1’, LMS invokes the ‘**add\_user**’ function as implemented in books. The system prompts the user to enter User name, password and password confirmation. Upon successful verification, system writes the user credentials in ‘**users.dat**’ binary file. System displays appropriate error message if the user fails to confirm the password.

**Lon In:** If user enters the option as ‘2’, LMS invokes ‘**login**’ function as implemented in books. As part of login, system asks the user to enter user name and password. System verifies the user credentials against information stored in **users.dat** file. System displays appropriate error message in case credential verification fails. Upon successful verification of user credentials, system displays the following options depending on whether the system is being used by an ‘Admin’ user or ‘Other’ users (other than Admin).

The ‘Admin’ user gets to see the following options:

1. **Add Books**
2. **Delete Books**
3. **View Books**
4. **Exit**

‘Other’ users (other than Admin) get to see the following options:

1. **View Books**
2. **Issue Book**
3. **Return Book**
4. **Exit**

The flow of control of application functionalities of different categories of users are described below:

**Application flow details for the Admin User:**

**Add Books**: If the Admin enterstheoption as ‘1’, LMS invokes the ‘**add\_books’** function as implemented in books. The system prompts the user to enter the unique book serial number, book name, author name and writes those information in ‘book\_details.csv’ file along with availability status as ‘Yes’.

**Delete Books**: If the Admin enterstheoption as ‘2’, LMS invokes the ‘**delete\_books’** function as implemented in books. The system prompts the Admin user to enter book serial number and book name to be deleted. The system validates the book serial number and book name and upon successful validation system deletes the requested book from the book\_details.csv file and displays confirmation message to the user. If the requested book does not exist, system displays error message.

**View Books**: If the Admin enterstheoption as ‘3’, LMS invokes the ‘**viewbook’** function as implemented in books. The system further allows users to opt for viewing all books and viewing issued books.

If user chooses to ‘**view all books**’, the system retrieves information from **book\_details.csv** file and displays those on the screen.

If user chooses to ‘**view issued books**’, the system retrieves information from **issued.csv** file and displays those on the screen.

**Application flow details for the Other Users (**other than Admin)**:**

**View Books**: If the user enterstheoption as ‘3’, LMS invokes the ‘**viewbook’** function as implemented in books. The View Books functionalities are same for other ‘Admin’ and ‘Other’ users as already mentioned above.

**Issue Book**: If the user enterstheoption as ‘4’, LMS invokes the ‘**issue\_books’** function as implemented in books. The system prompts the user to enter specific book serial number and book name to be issued. The system validates the requested info against entries in book\_details.csv. Upon successful validation, system writes information into issued.csv file along with book details, issue date, return date and displays confirmation message for issuing the book.

**Return Book**: If the user enterstheoption as ‘5’, LMS invokes the ‘**return\_book’** function as implemented in “book.py”. The system prompts the user to enter specific book serial number and book name to be returned. The system validates the requested info against entries in issued.csv. Upon successful validation, system removes the entry from **issued.csv** file and updates the **availability status to ‘Yes’** in **book\_details.csv** file.

**Exit (from LMS):** If the Admin user or ‘Other’ user opts for entering ‘0’, system displays ‘Thank You’ message to the user for using the LMS

**Scope of project:**

It may help collecting perfect management in details. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of passed year perfectly and vividly. It will also reduce the cost of collecting the management & collection procedure will go on smoothly.

Some other noticeable features of this management system include:

* Good user interface
* Easy to operate
* Has ability to meet user’s requirement
* The system generates types of information that can be used for various purposes.
* Files can be easily exported easily
* Personal information of users including username and password are protected.

**BIBLIOGRAPHY AND REFERENCES**

1. ***Computer science With Python -  Class XII By : Sumita Arora***
2. [***https://pythonworld.in/practical-project/project-list/***](https://pythonworld.in/practical-project/project-list/)
3. [***https://www.youtube.com***](https://www.youtube.com)