Computer Science Project

on

Library Management System

Debayan Sutradhar

12 M

DPS Ruby Park, Kolkata

Certification

This is to certify that the project entitled:

“SnakeBrary - A Library Management System”

is the original work of Debayan Sutradhar, 12 M (2020/14275), DPS Ruby Park School, Kolkata. This system was developed as CBSE Class 12 Project.

______________	______________
Internal Examiner	External Examiner

Acknowledgement

I would like to thank my parents for funding my education and providing computer to develop this project on and also internet services. I am also grateful to the developers of the Python programming language, and the respective owners/developers of all the 3rd party services and libraries I have used for this project.

Synopsis

Objective

The goal of the system is to be a simple, free, open-source, cross-platform, user-friendly, small, light-weight and scalable Library Management System. The source code for this program is available on GitHub, and this system is licensed under the GNU GPLv3 License, which allows others to improve and even make their own custom versions.

- A user login system that makes it easier to organise and administrate the system.
- A powerful search system to search for a book by its name, author name, ISBN or even genre.
- Automates and digitises the entire process of issuing/returning books and records all information digitially in a database.
- Rate already read books, that helps others to know general idea about the book.
- Lightweight system that can be run on cheap and low end hardware rather than a full computer system, hence reducing costs.
- Free and Opensource alternative to existing library management system.
- Cross platform alternative that can be run on a variety of hardware and Operating Systems.
- Easy to use and intuitive user system while maintaining the same amount of features as a regular library management system.

The system is based on the principal of users. Users are further divided into 3 different types:

1. Normal: This user is the most basic type of user and is consists of the average Library user. They can only issue, return, rate and search for books. They can also edit their own user information.
2. Administrator: This user can create, edit or delete users and books. They cannot delete the 'Master' or other Administrators. They can also do anything that a 'Normal' user can do.
3. Master: This user is created during the intial setup of the software. There can only be one master user. This type of user can create, edit or delete any user or book from the system. They can also reset the entire software, which will delete the System database from the SQL Server.

Input and Output of the Proposed System

The inputs and outputs of the system vary based on the privilege of the User logged in. It depends on the privilege of the user. The inputs and outputs have been described for each user account privilege below.

1. Normal User:
   a. Input:
      i. Book Name, Author, genre or ISBN for searching a book.
      ii. Rate a returned book.
      iii. Edit their own user information.
      iv. Return a book.
      v. Issue a book if available.
      vi. Change theme and accent colour of the application.
   b. Output:
      i. Table of all books available.
      i. The user can click on a book. and then get details like the Name, ISBN, Genre, and About & Cover Photo (if exists). They can also look at the average rating of the book rated by other readers.
      ii. The user can see their own information in Settings > About Tab.
2. Administrator:
   a. Input (Includes the points of a user with Normal Privilege):
      i. Create/Edit/Delete users with Normal privilege.
      ii. Add/Edit/Delete books.
      iii. Return a book from any user (except user of the same privilege and Master).
      iv. Delete rating of a book by users of normal user.
      v. Books can be made unavailable temporarily for users
      vi. Normal User accounts can be disabled.
   b. Output (Includes the points of a user with Normal Privilege):
      i. See list of all users.
      ii. View information of any user like Name, username, Date/Time Added. Password and Password Hint of Users of the same privilege and the master cannot be viewed.
      iii. View list of people who reviewed a book.
      iv. View the issue/return date and time of all the holders of a book in order.

v. User accounts with any privilege can be disabled.

vi. View the current holder of a particular book.

3. Master:

a. Input (Includes the points of a user with Normal and Administrator Privilege):

i. Create/Edit/Delete accounts with any privilege.

ii. Return a book from any user of any privilege.

iii. Delete System database and local database by reset button.

iv. Delete rating of a book by any user.

b. Output (Includes the points of a user with Normal and Administrator Privilege):

i. View user information including password and password hint of every account.

Functions or Features of Proposed System

1. User Account System: The system works on the main principal of a user-account system. This makes it easier to maintain the system, especially for bigger environments. Accounts can also be disabled and certain books can also be made available by the administrator or the master user.
2. Book rating System: Each book can be scored out of 5 points by any user who has read it. This helps other users know about how the book is generally perceived by the audience.
3. Powerful search interface: The user can search for a book by its name, author, ISBN or even its genre. An administrator/master user can also search for a user by their username or name.
4. Modern and clean UI: This system uses PySide6 for GUI, and adheres to Google's Material Design Language, which makes it far cleaner and modern than a standard Tkinter application.
5. Intuitive and User Friendly Interface: No manual or help is required to use the system. Everything is self explanatory and designed with simplicty in mind. Each book can have an 'About' section that the user can go through before issuing the book. Books also support addition of a cover photo and users support addition of profile picture.
6. Cheap, Portable and Lightweight: This system is cross-platform, and can be run on a variety of different systems, even on low-end, cheap platforms like Raspberry Pi. This could significantly bring costs down since a lower end single computer could be used to run this rather than a traditional computer.

Front-end and Back-end

- Front-end: PySide6 (Qt Framework Bindings for Python)
- Back-end:
    - MySQL for System Database
    - SQLite3 for storing connection details to MySQL Server (username, password, hostname and port)

Requirements

- OS: Microsoft Windows 7/8/8.1/10, Linux, Apple MacOS X
- RAM: 200 MB
- Processor: Any x86, x86_64 or ARM processor
- Storage: 50 MB (Not including MySQL Server)
- Python: 3.9+
- 3rd Party Python modules
    - PySide6 (PySide6)
    - Qt Material Stylesheet (qt-material)
    - Python MySQL Connector (mysql-connector-python)

Scope and Limitations of the Project

This program is designed with simplicity, scalability and poratability in mind. Hence it can be deployed to multiple computers of various platforms and connect to the same MySQL Server. It is also very lightweight which is why it can also run on cheap, low end and low footprint devices like Single Board Computers (Raspberry Pi, etc.).

However, certain limitations remain:

- This program cannot be ported directly to micro controllers which could have potentially brought down the cost of the entire system. This is due to the fact that this system was developed in Python which is not as flexible as languages like C/C++. Hence, an Operating System is required, which this program runs on top of, requiring more hardware power.
- This program cannot be run on mobile platforms like Android or iOS due to limitation of the Python platform and Qt Framework.

Bibliography

- PySide6 Documentation
- Python MySQL Tutorial
- Stack Overflow