Library Software

1. Introduction

The main idea of the project is to design a fully functional software that can be directly and conveniently used by any Library. For the software to be functional for any Library, it consists of a sample database of over 10,000 books. The database consists of the names of the books, their authors, their year of release and also their user ratings out of 5. This database can however be customized as per the Library using it and a Library can add their own database of books as well through our help.

We have tried to incorporate majority of the functions that a Library would expect from a software. For all the Libraries that still use a manual mechanism to store their member details, their data of books in stock, books issued and books returned, this software is no less than a blessing. What this software excels in, is the fact the each member is notified about his/her membership details, due date of each book taken, and the penalty imposed if the return is beyond the due date, at a personal level. Each member personally receives an email on their registered email id and a unique membership number is given to each member for ease of use.

Why this software over others, is a simple answer. We believe that our software is light enough for any machine yet featured enough that nothing more would be required. This, along with the promise of constant backend support for our clients and the future scope of this software makes this a compelling choice.

2. Design

This software has been completely designed in Python language. For database purposes, MySQL has been used. For commercial purposes we do not intend to give the detailed code in this document. However we do intend to share all the modules used in the programming of this project. Underneath are all the required details with links to their official documentation.

1. MySQL
2. PyQt5
3. datetime
4. smtplib

2.1 MySQL

MySQL is an open-source relational database management system(RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

We have used MySQL for creation of database. We have accessed and modified data using Queries.

Documentation for MySQL v5.7: https://dev.mysql.com/doc/refman/5.7/en/

2.2 PyQt5

Qt is set of cross-platform C++ libraries that implement high-level APIs for accessing many aspects of modern desktop and mobile systems. These include location and positioning services, multimedia, NFC and Bluetooth connectivity, a Chromium based web browser, as well as traditional UI development.

PyQt5 is a comprehensive set of Python bindings for Qt v5. It is implemented as more than 35 extension modules and enables Python to be used as an alternative application development language to C++ on all supported platforms including iOS and Android.

PyQt5 may also be embedded in C++ based applications to allow users of those applications to configure or enhance the functionality of those applications.

We have used PyQt5 for the GUI of our software.

Documentation for PyQt5: http://pyqt.sourceforge.net/Docs/PyQt5/

2.3 datetime

The datetime module supplies classes for manipulating dates and times in both simple and complex ways. While date and time arithmetic is supported, the focus of the implementation is on efficient attribute extraction for output formatting and manipulation.

We have used this module for the software to store member registration dates, issue dates and due dates.

Documentation for datetime module:
https://docs.python.org/3/library/datetime.html

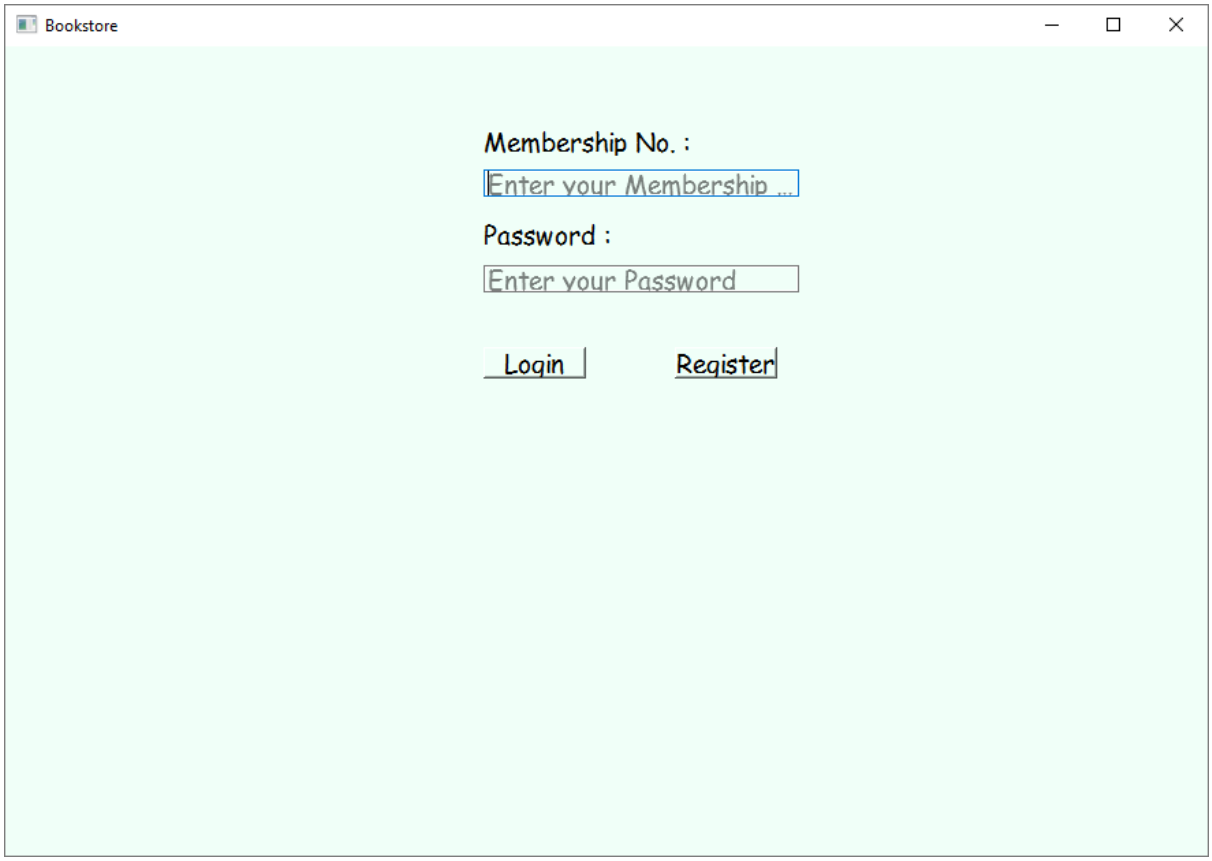
2.4 smtplib

The smtplib module defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP or ESMTP listener daemon. For details of SMTP and ESMTP operation, consult RFC 821 (Simple Mail Transfer Protocol) and RFC 1869 (SMTP Service Extensions).

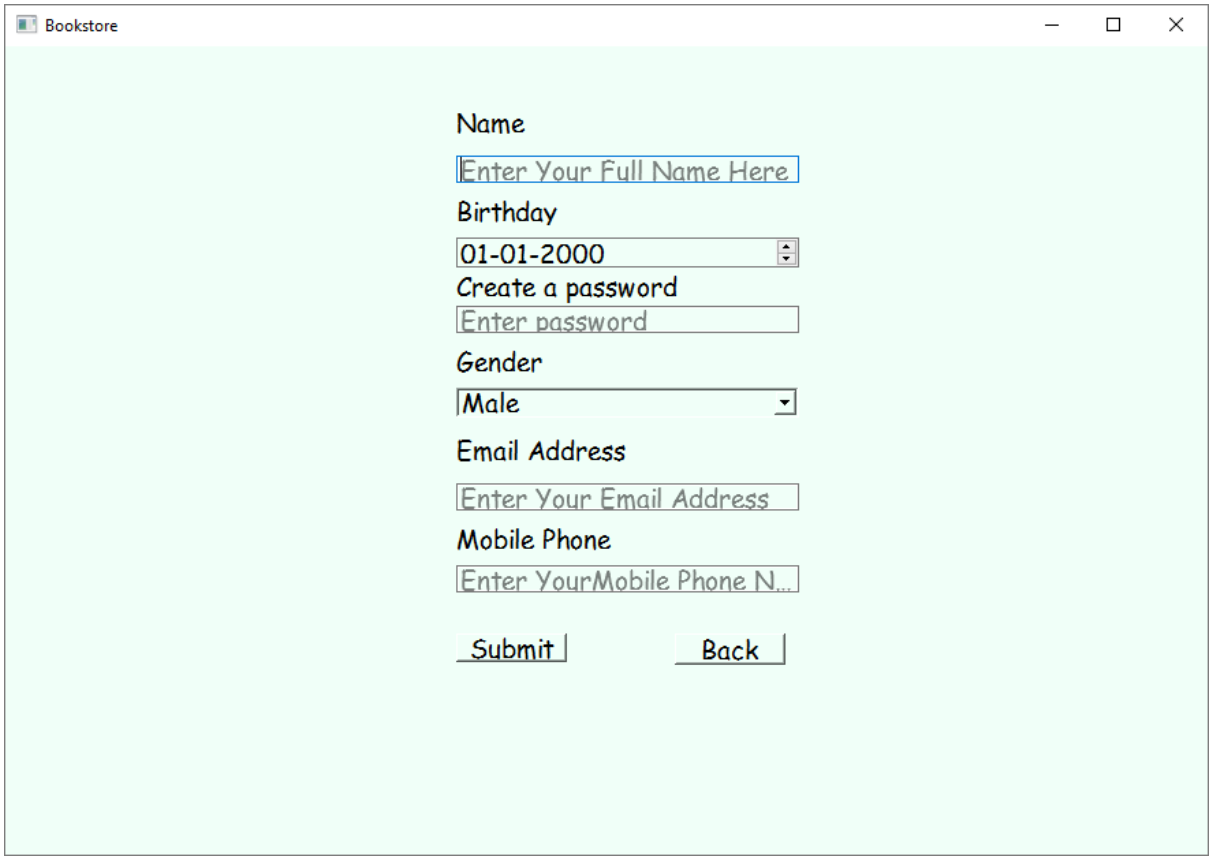
We have used this module to code a feature wherein each member receives personalized details of his/her membership, books issued, and penalty details on his/her registered email id.

Documentation of smtplib module:
https://docs.python.org/3/library/smtplib.html

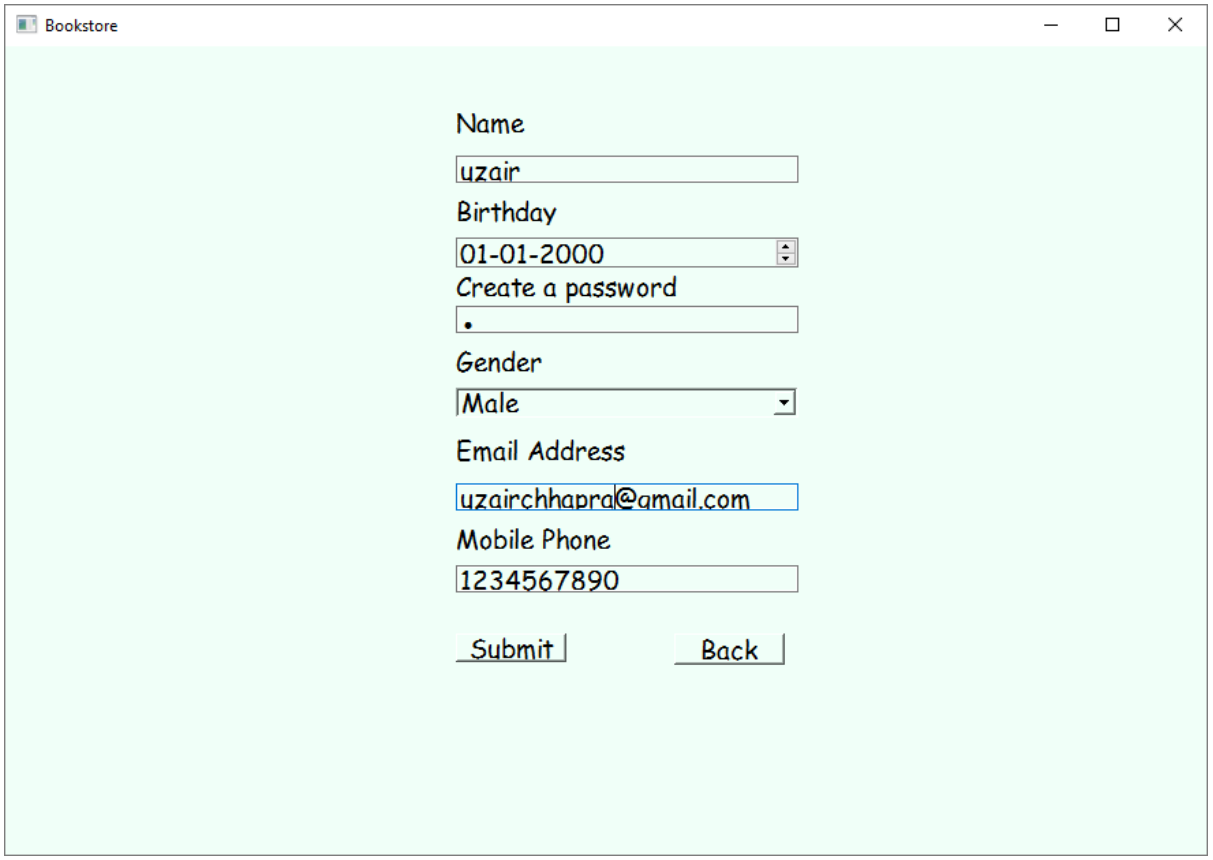
3. Execution



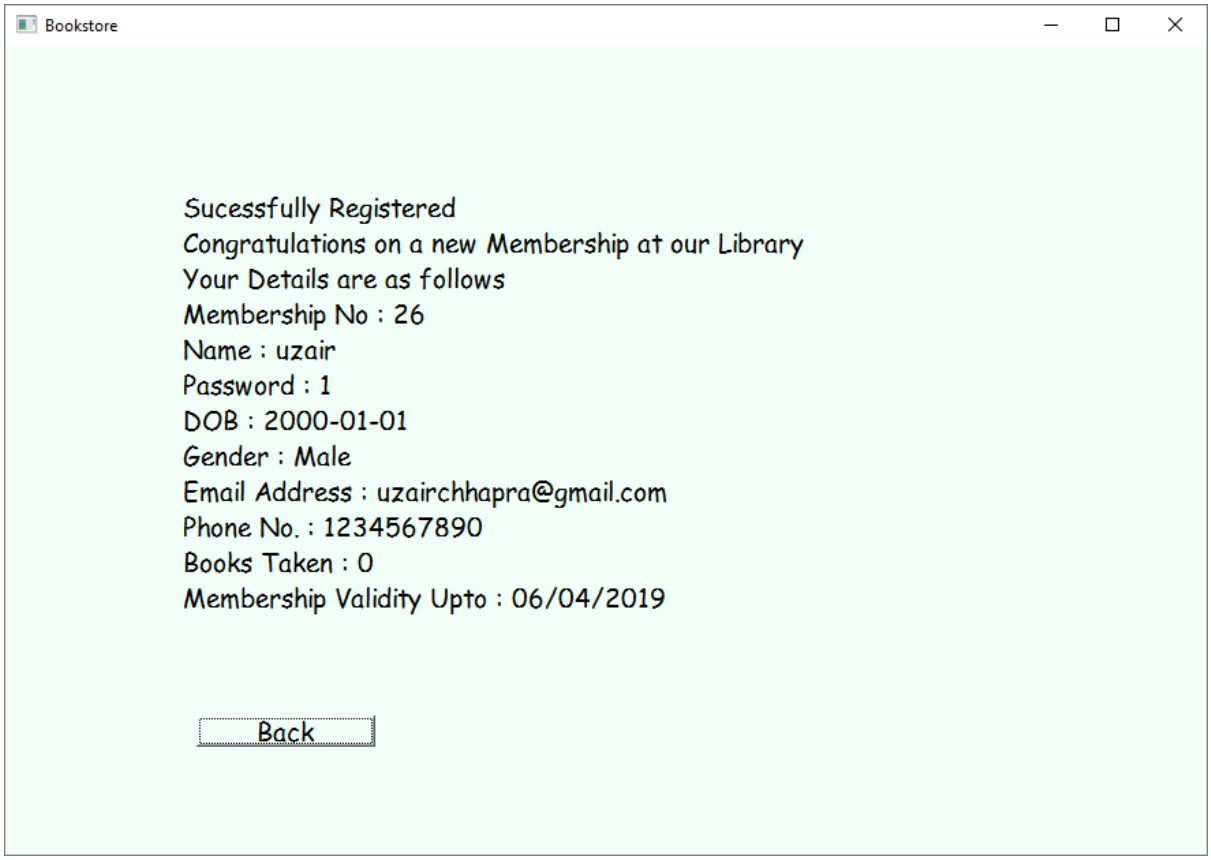
3.1 Login Page



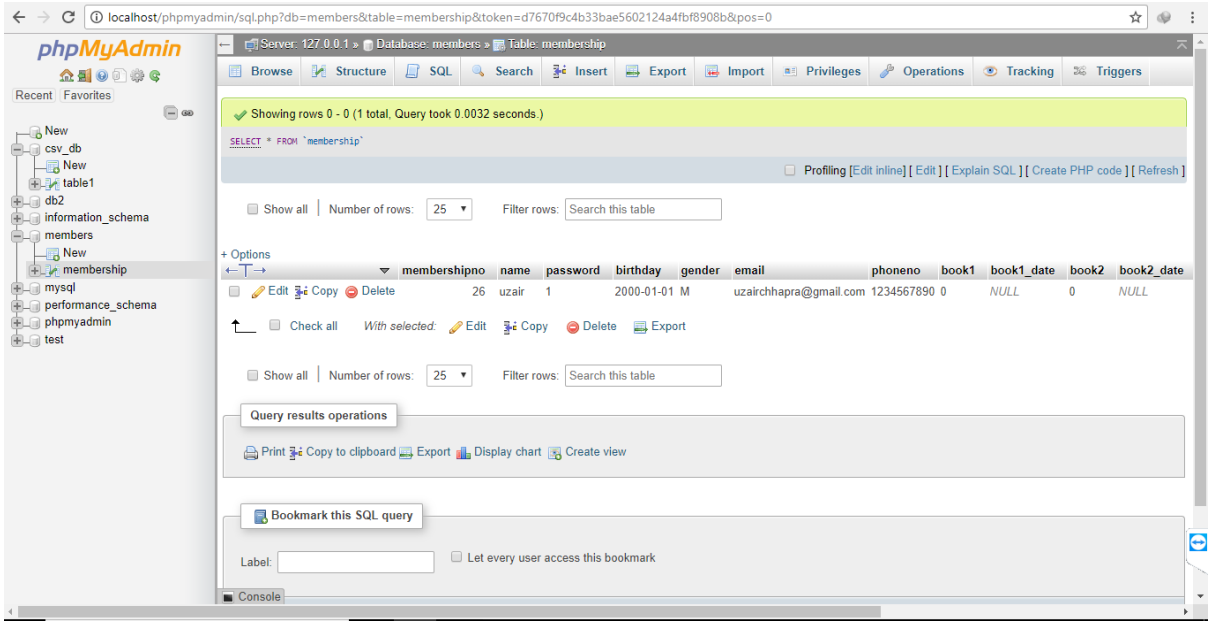
3.2 Registration Page



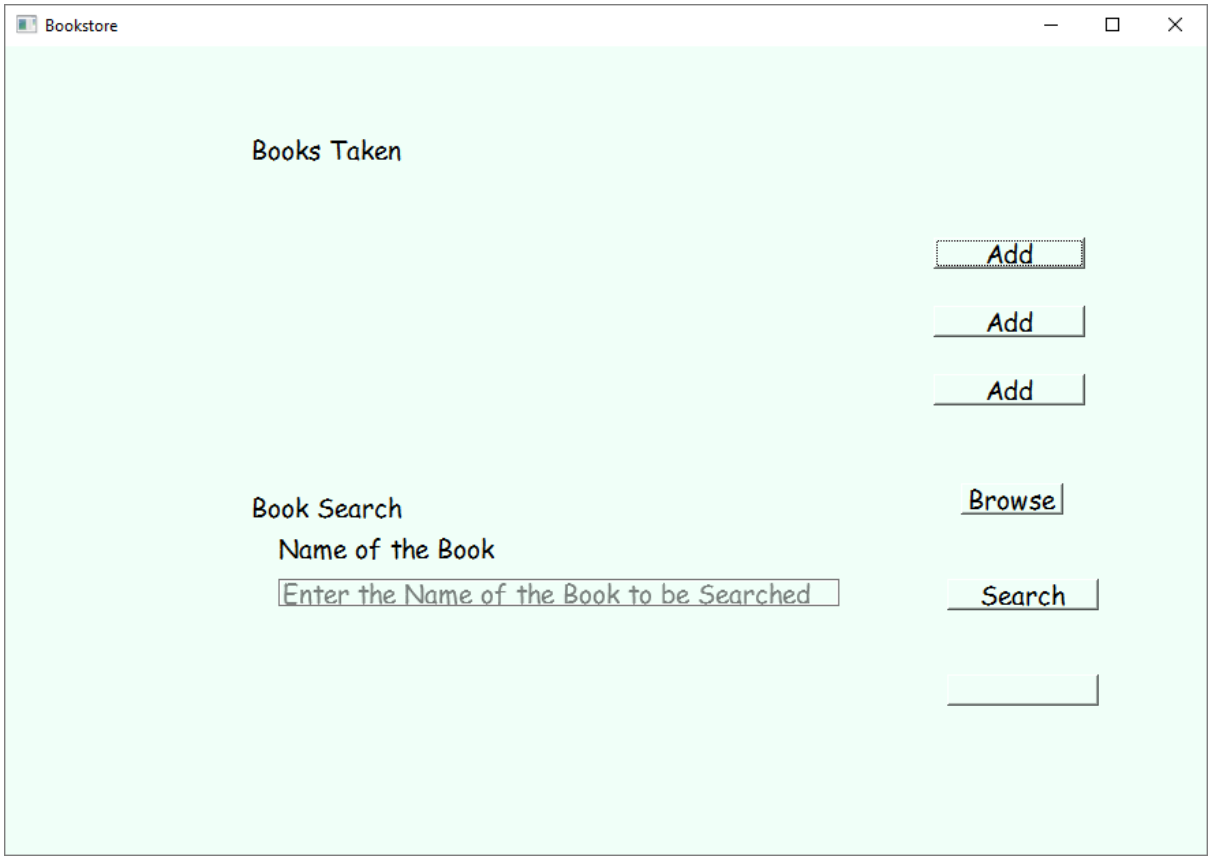
3.3 Filled Registration Page



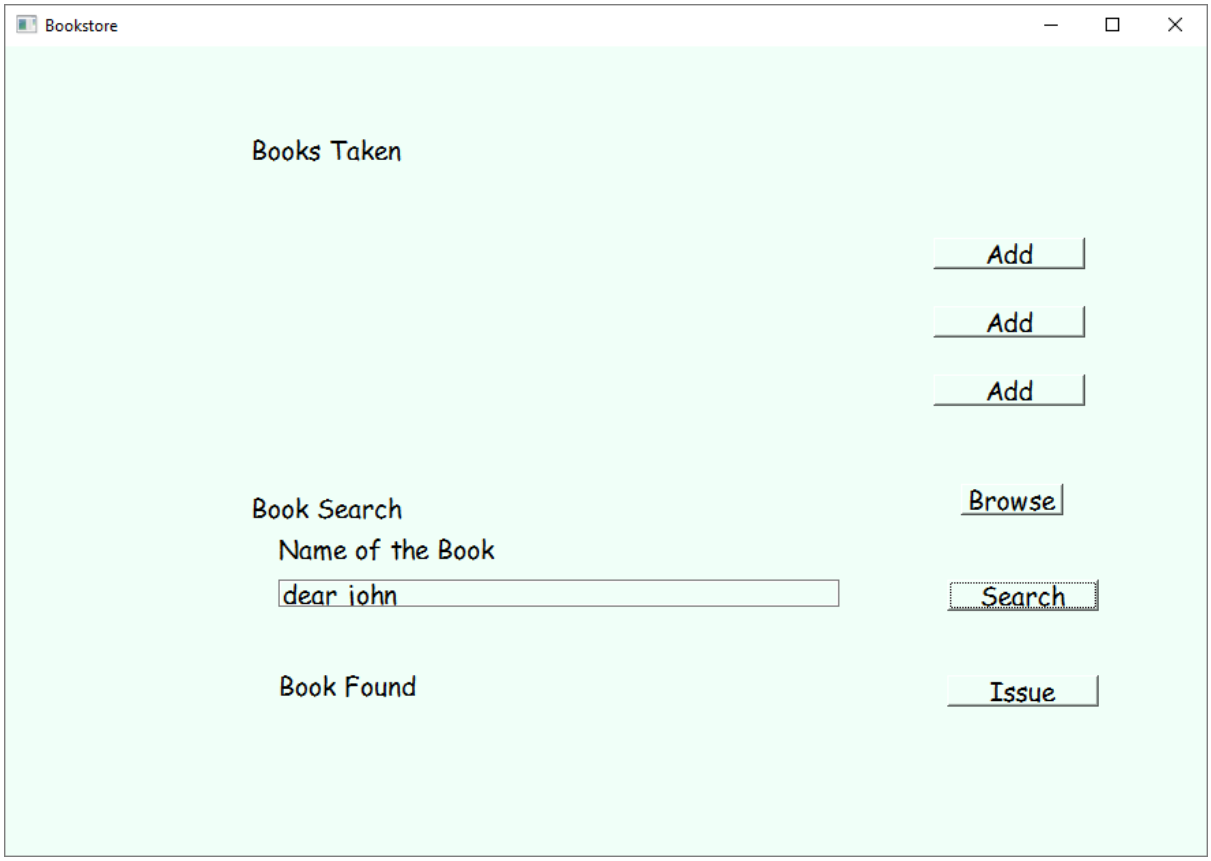
3.4 Successfully Registered Page



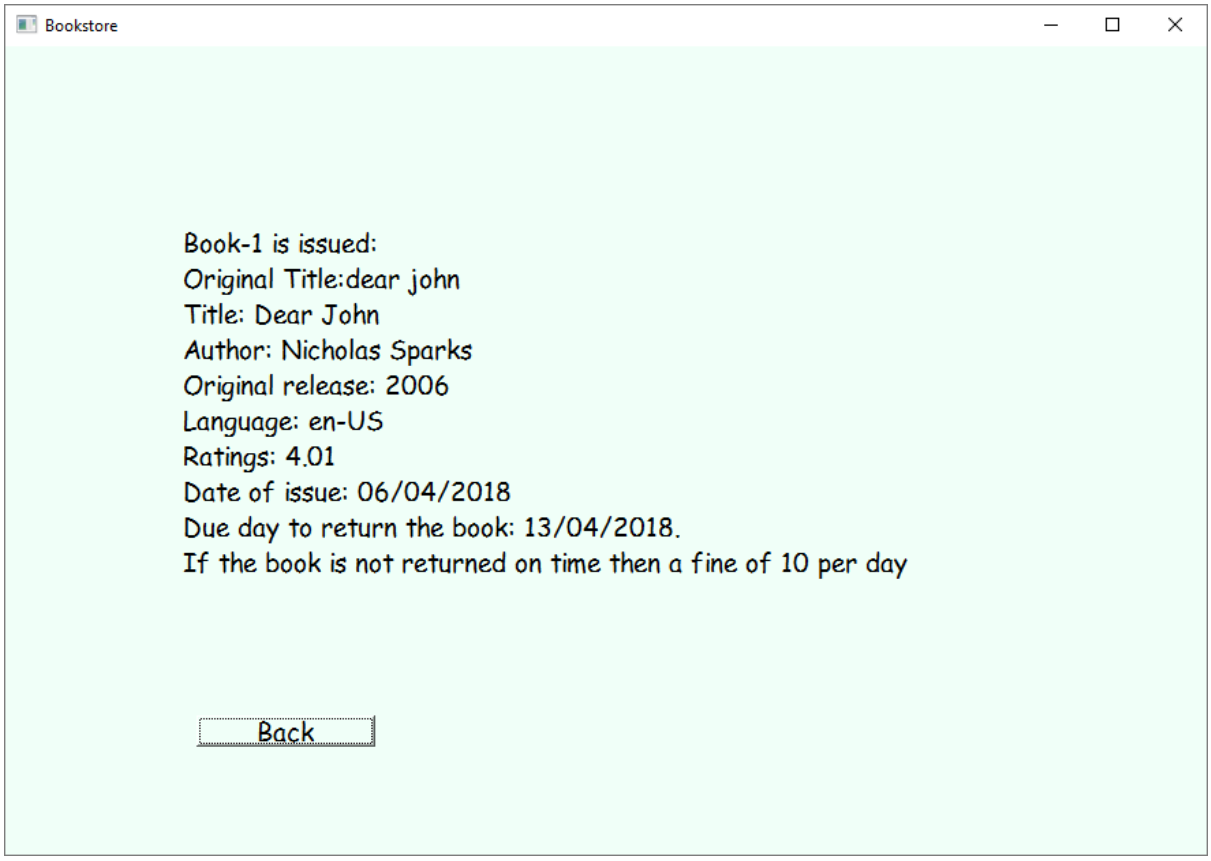
3.5 Members Database Page



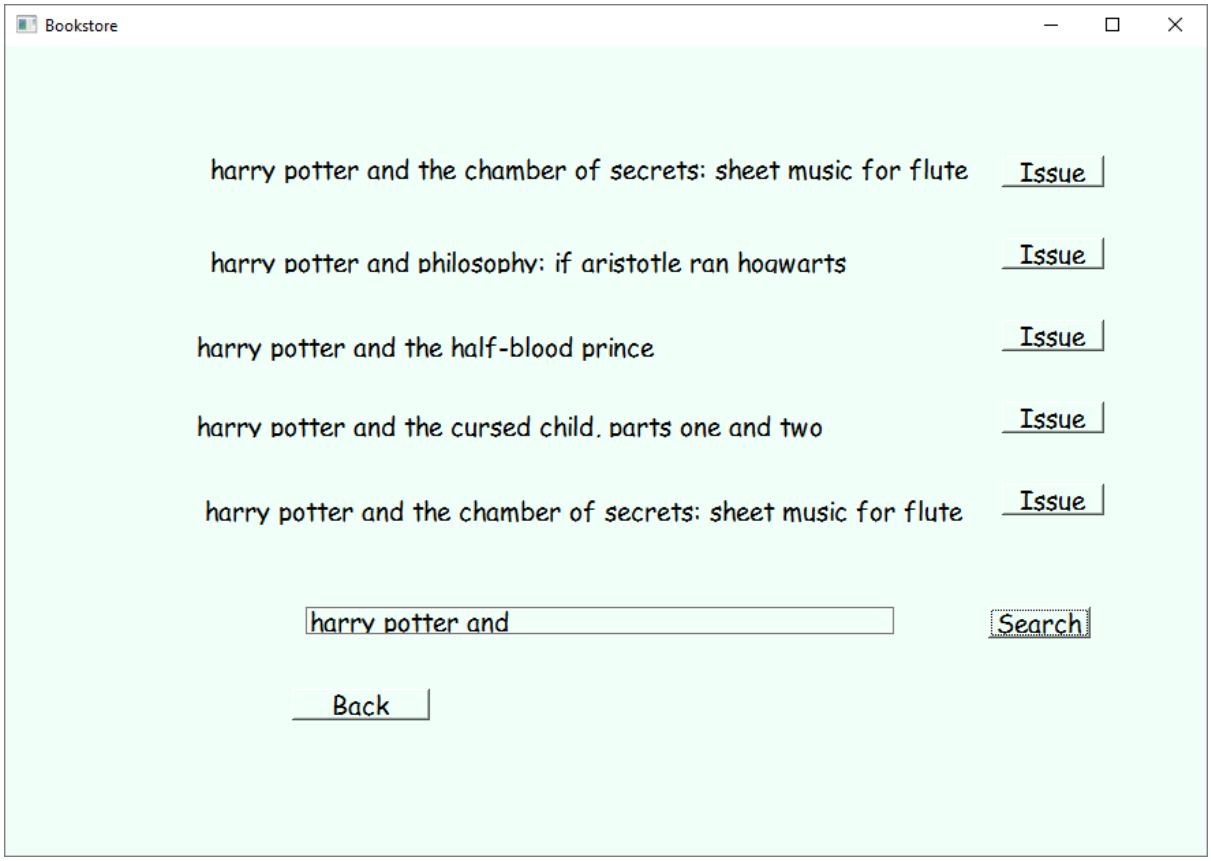
3.6 Logged In Page



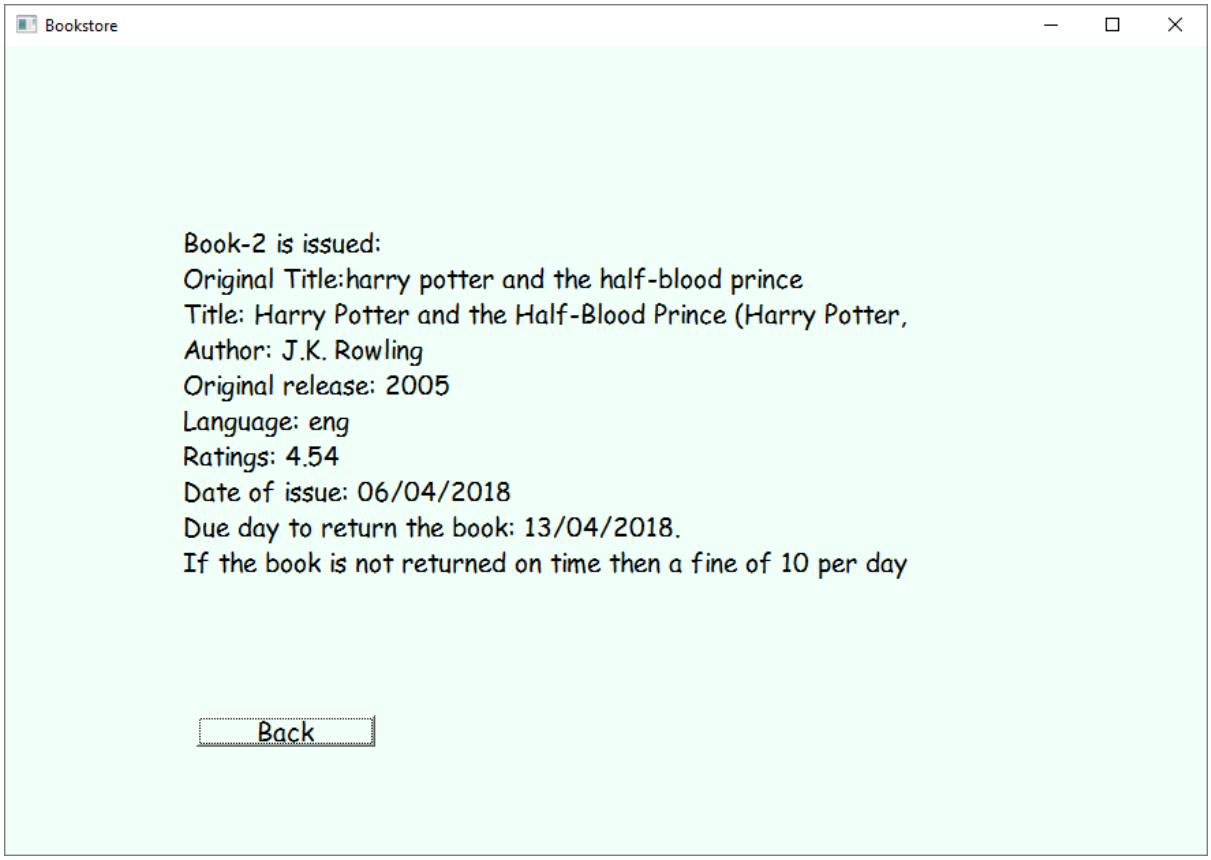
3.7 Book1 Found Page



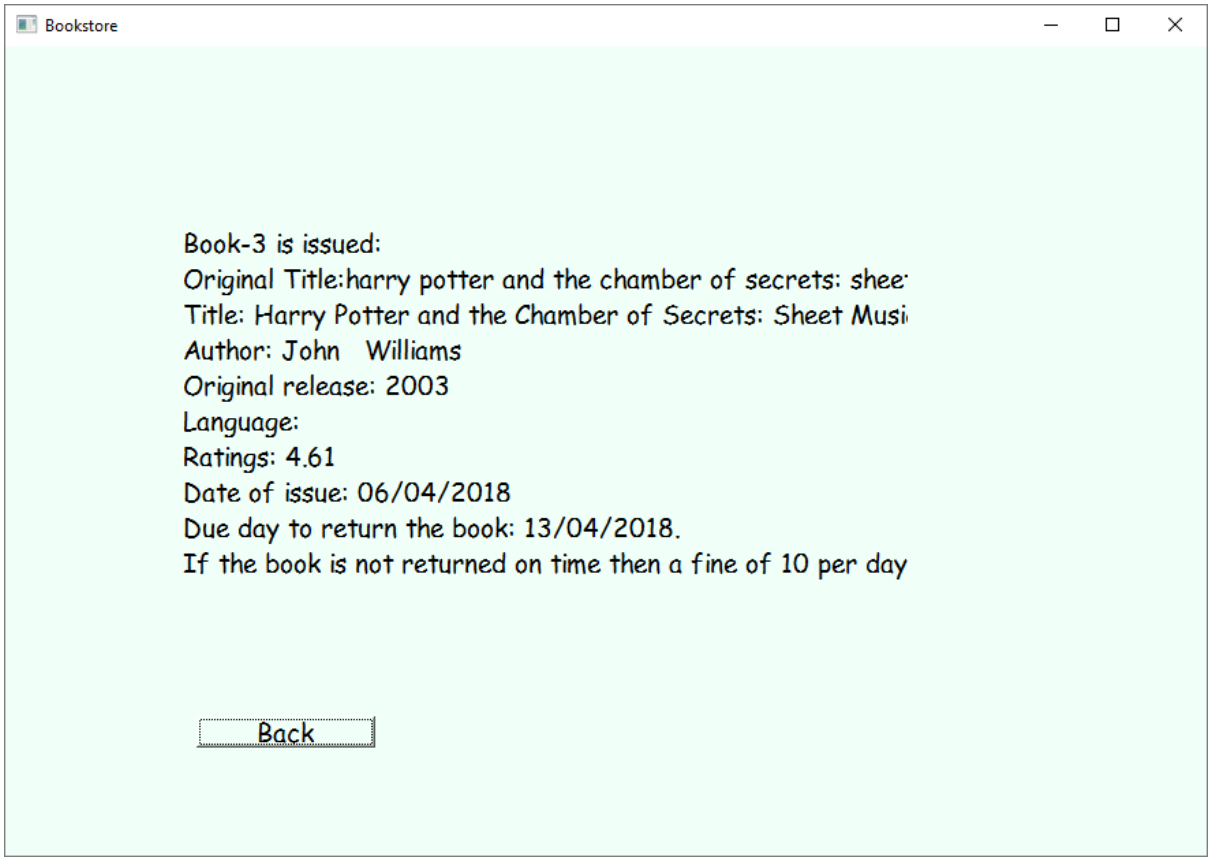
3.8 Book1 Issued Page



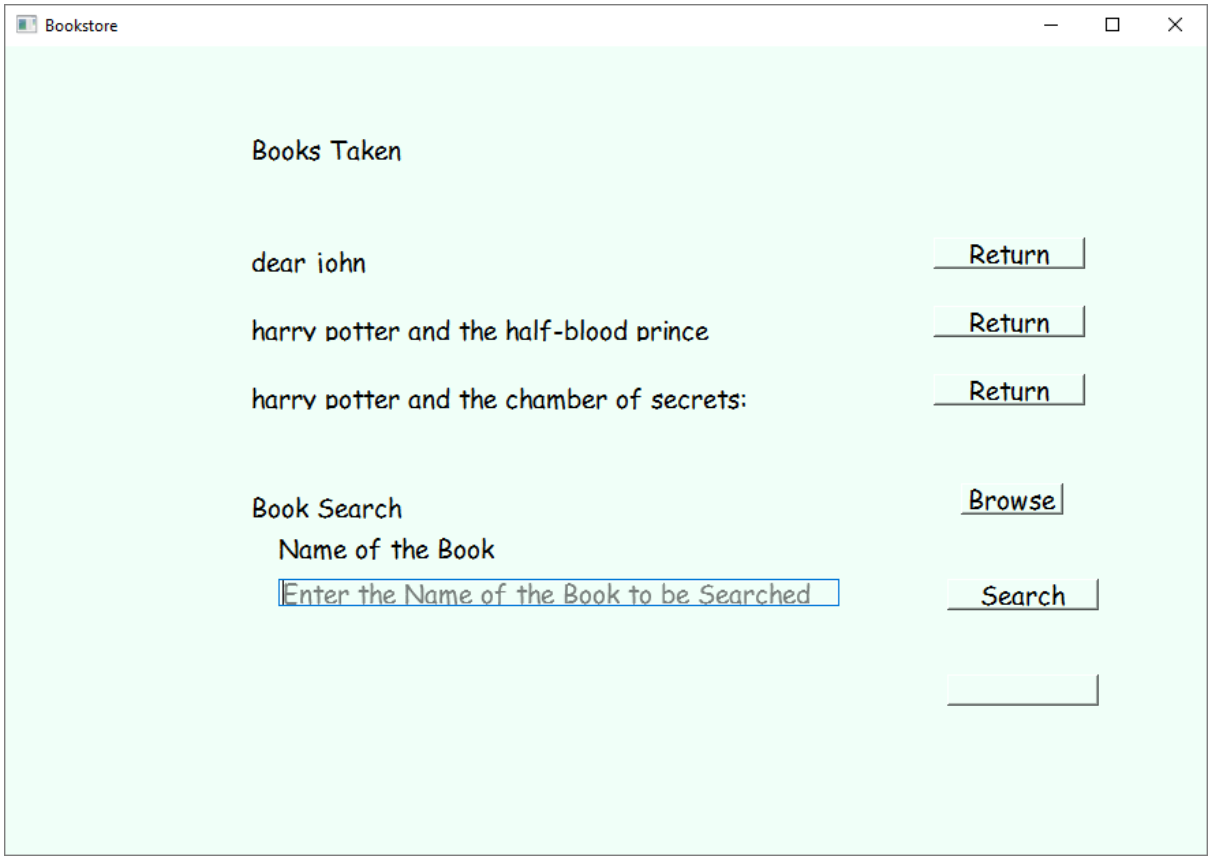
3.9 Browse Books with Similar Names



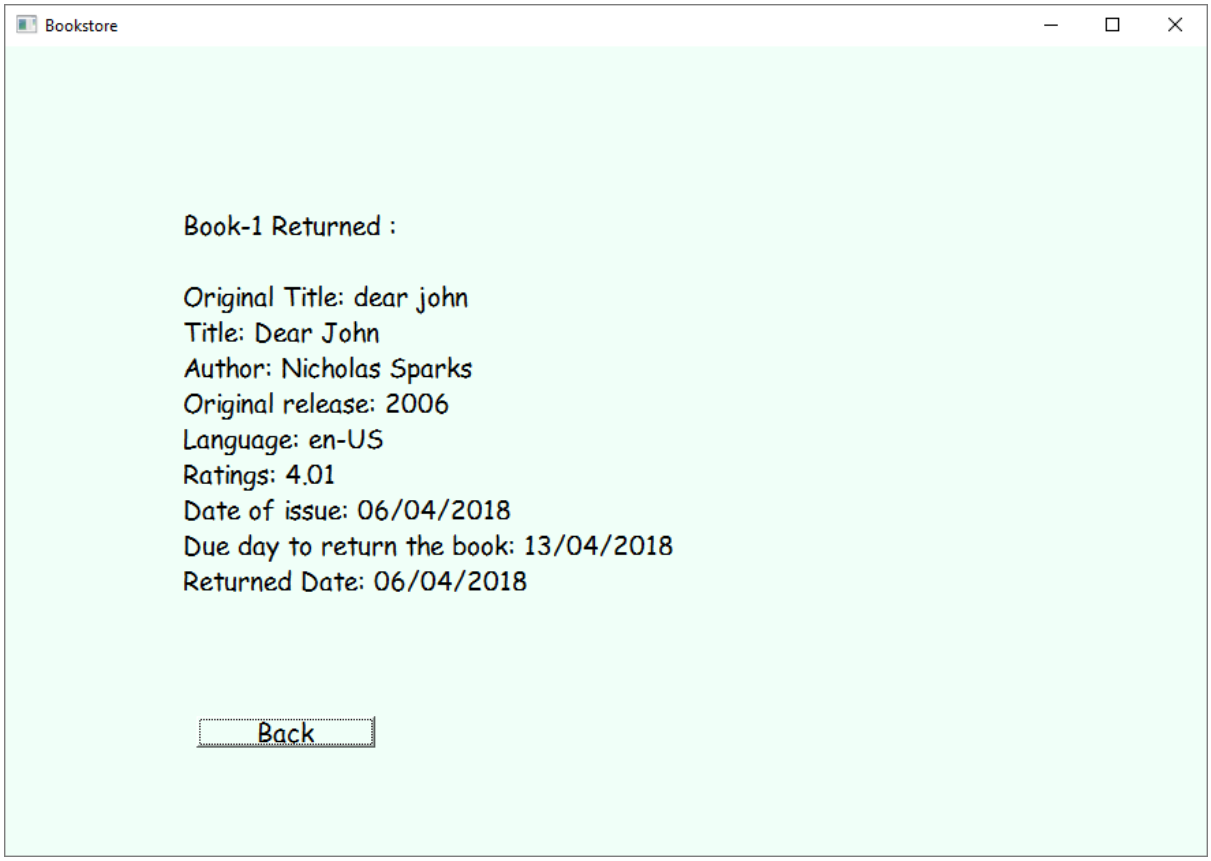
3.10 Book2 Issued Page



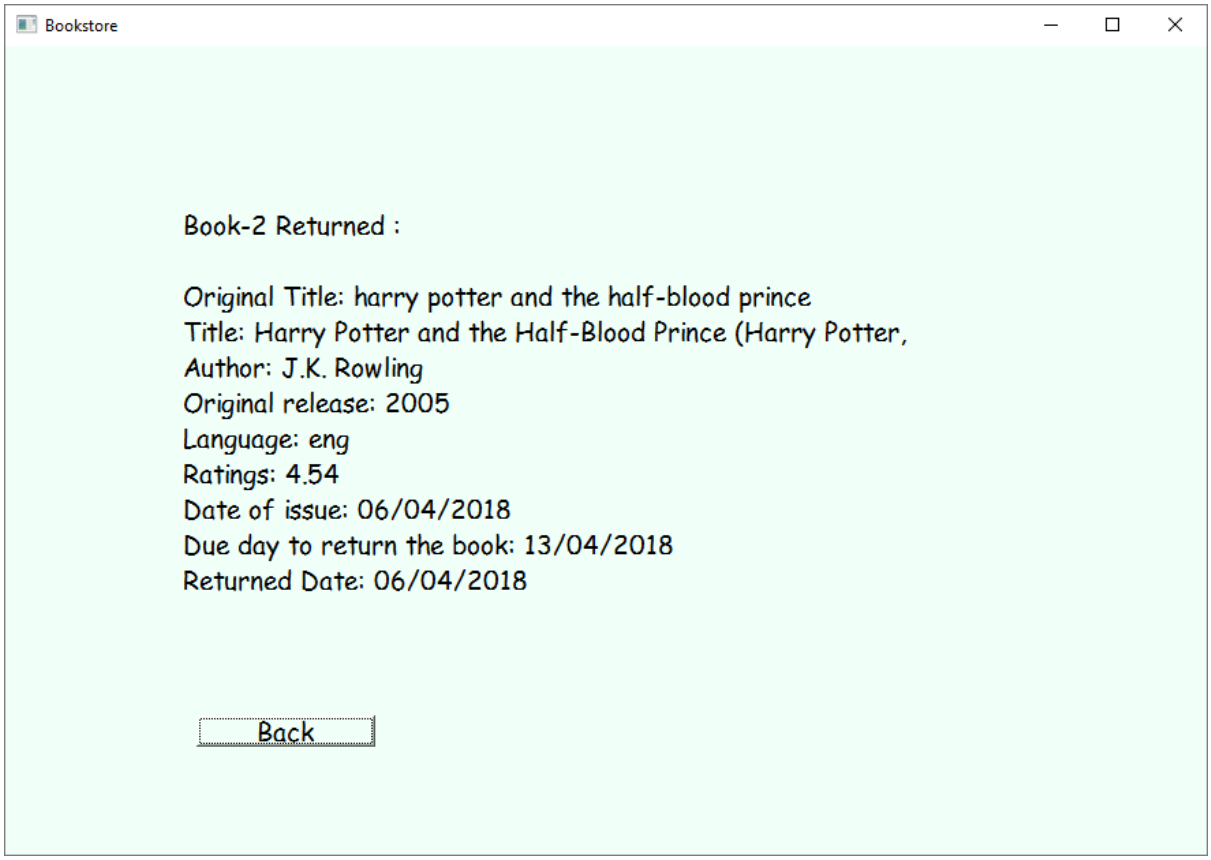
3.11 Book3 Issued Page



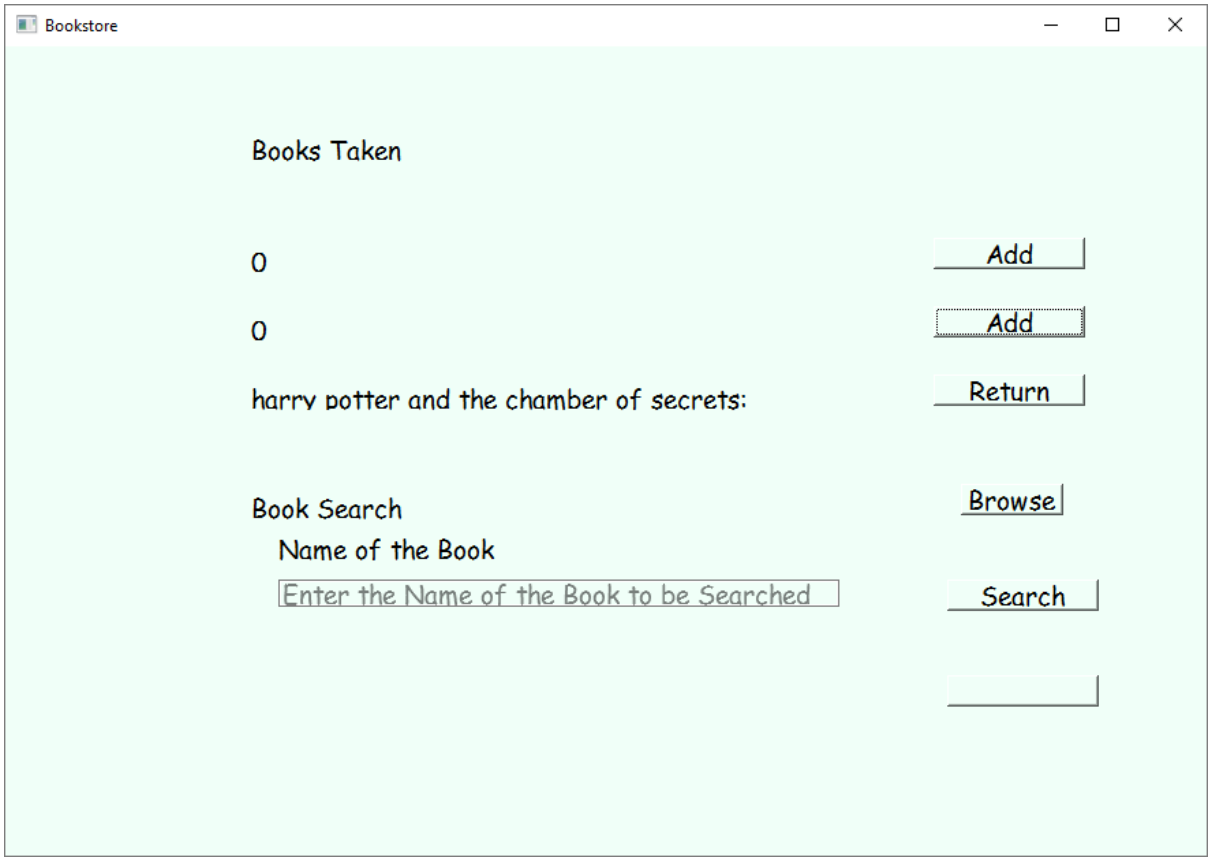
3.12 All 3 books Issued Page



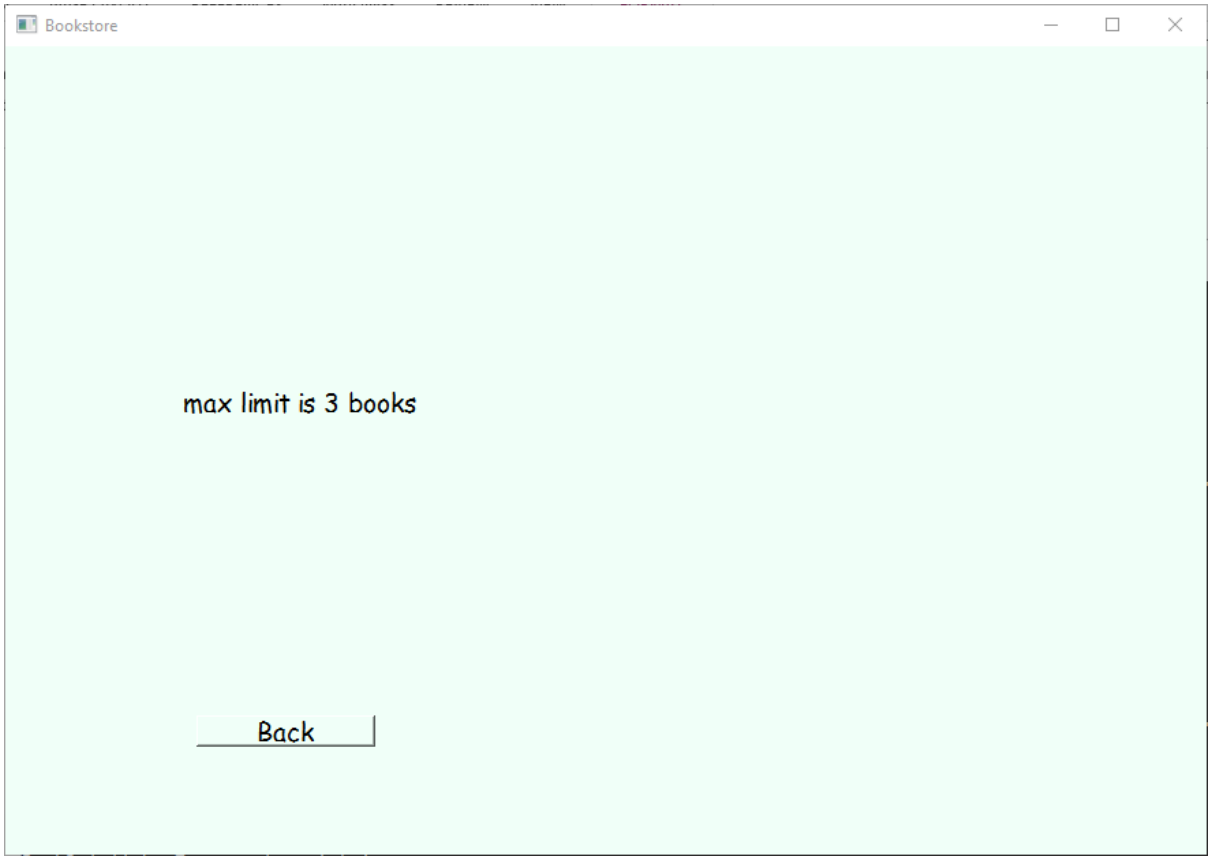
3.13 Book1 Returned Page



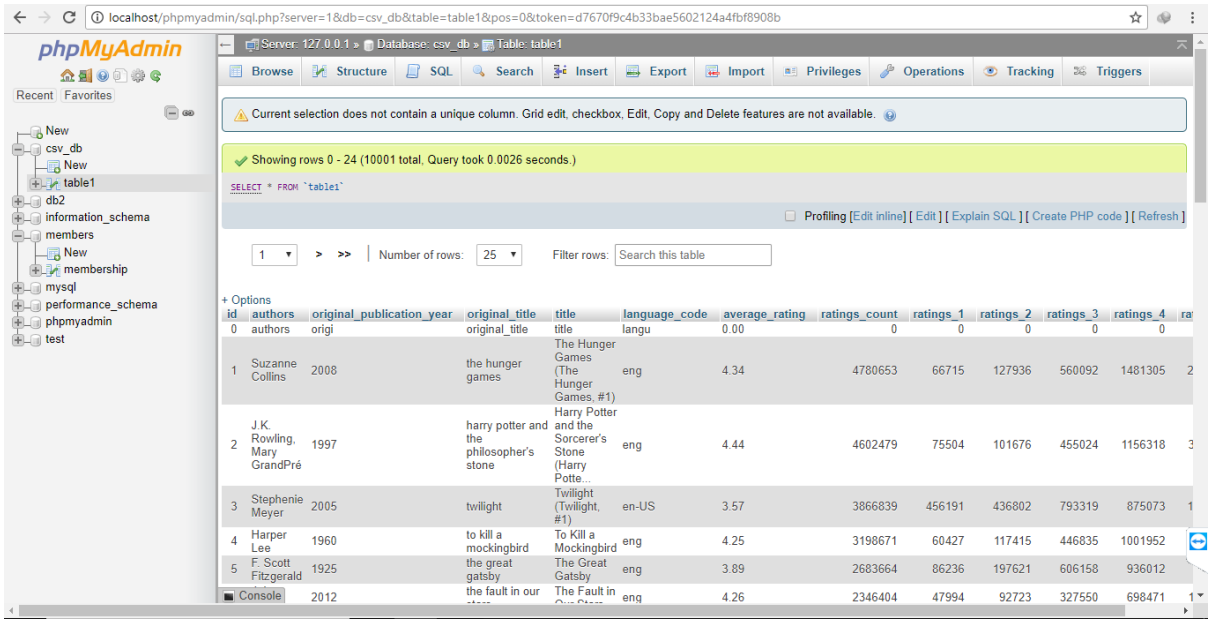
3.14 Book2 Returned Page



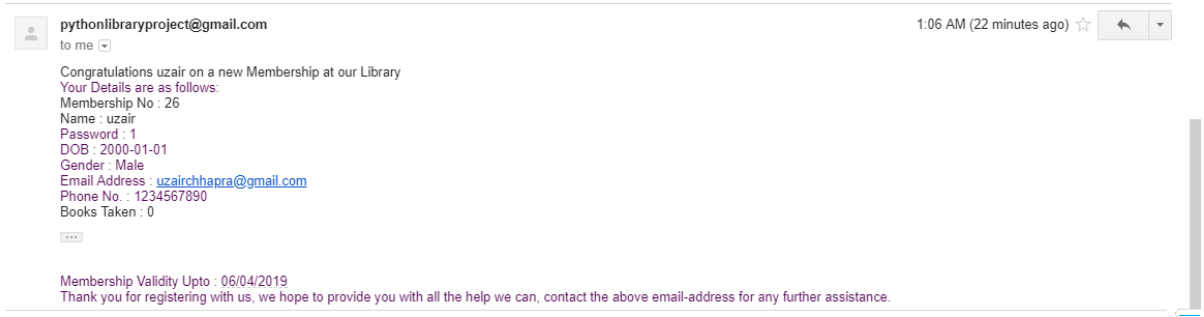
3.15 Page showing 2 books returned



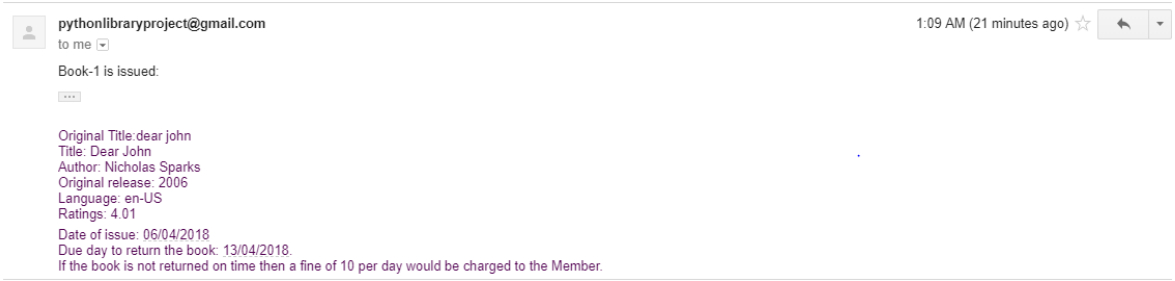
3.16 Max Limit Page



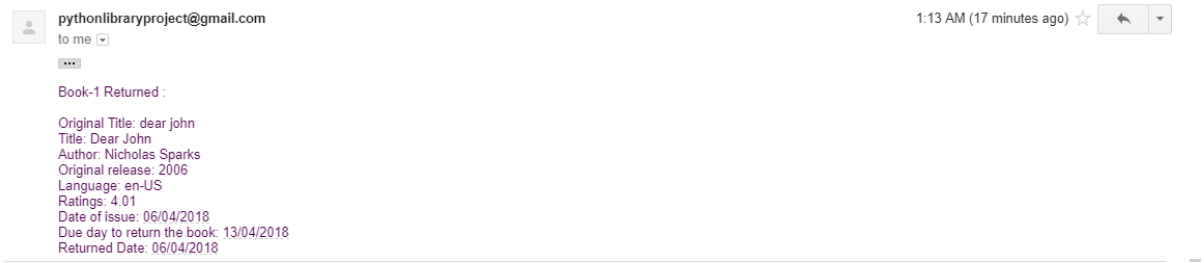
3.17 Books Database



3.18 Email of Membership Details



3.19 Email of Book Issued



3.20 Email of Book Returned

4. Future Scope

Certainly, even though we have tried our level best to include all possible requirements that a Library could seek from such a software, we know that there will improvement required as per the client demand.

Currently, the software is capable enough to handle all basic functions of a Library, but enhanced features for members like the option to select books based on popularity ratings, receiving SMS updates, ability to connect with friends and know what books they have currently issued are not available. Also the GUI implemented in our software is very minimal which may or may not be liked by many.

The above mentioned limitations are definitely not a necessity for any Library. But we understand how important user experience is. And so, we promise to add the above mentioned features if there are clients interested and willing to use our software.

GitHub Link to our project: https://github.com/adityachavan198/Library-app