CIS 227 Assignment 2

Assignment Details

Use at least one class

Create a program to read inventory from a JSON, or CSV file

Create a “login” process. -Use clear text passwords and a CSV file to store the information.

Prompt the user to search the “database” for titles and display the following information:

* Book Title
* Book Author
* Publisher
* Book Publication Year
* Book Description

Team Roles

Lead Programmer – **Fernando Alvarez**

UX/UI Programmer – **Christian McQueen**

Functional Programmer – **Skyler Metzger**

Program – 70

UX/UI – 35

Function - 35

Documentation – 30

Total Possible Points – 100

**Version 1.1.0**

| REVISION HISTORY | | | |
| --- | --- | --- | --- |
| DATE | VERSION | DESCRIPTION | AUTHOR |
| 1/11/2023 | 1.0.0 | A program that calculates the hypotenuse of a triangle, the area of a trapezoid, and the volume of a rectangle. User input is required for each formula. | Christian McQueen |
| 1/18/2023 | 1.1.0 | Convert Assignment 0 to utilize a class structure using portable .h and .cpp files. The program also only exits on user demand. | Christian McQueen  Fernando Alvarez  Skyler Metzger |
| 1/25/2023 | 1.0.0 | Create a book inventory program that reads inventory from a JSON/CSV file. Include a login system using clear text passwords and store that information in a CSV file. Prompt the user to search the inventory for titles and output necessary information about them. | Christian McQueen  Fernando Alvarez  Skyler Metzger |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# INTRODUCTION

## PURPOSE

Identify and describe scope of product whose technical specifications are being documented and describe desired outcome.

**The scope of this product includes students or users of the database system in search of books within our inventory, and the administration that keeps the data updated as well as secure from those without credentials. The desired outcome is to support a user-friendly system for finding a book or books and provide relevant information for those books.**

## DOCUMENT CONVENTIONS

Describe any naming or structural conventions employed throughout document and how they benefit reader.

**Naming and structural conventions employed inside the program relate to the program’s functions. For instance, class variables ISBN, year, publisher, description, title, and author are used and crucial information regarding the search function to find the book the user is looking for.**

## REFERENCES

List any referenced document names or links.

https://cplusplus.com/doc/tutorial/files/

# DESCRIPTION

## FEATURES

List main features with brief description.

**Main features include the login system, which checks user credentials to allow for the database to be searched. The next feature is the database itself, which is combed for relevant information based on user input.**

## USER OVERVIEW

Define groups and describe user characteristics.

**Groups: The administration (at this time, this is just the group and the professor) with the admin login credentials, and the normal-clearance users accessing the database. This is assumed to be students in a library but could be any user at a library researching books.**

## ASSUMPTIONS / DEPENDENCIES

Detail all assumed factors (not known facts) that could potentially impact technical specifications set forth. Include external factors.

**As of this assignment, there is no feature to create a login for accessing the database. The one set of credentials being passed through the program is the login.csv username and password that is already determined. In this instance, the user (Professor Carmon and my group) has access to this database. However, the real world would want to account for a user that was given their credentials but may have lost one or more of them over time. Most user accounts have options for creating logins, updating logins, and recovering accounts. This program does not have any of those features currently.**

# SYSTEM FEATURES

## SYSTEM FEATURE 1

|  |  |
| --- | --- |
| **DESCRIPTION AND PRIORITY** | **Login process to protect data and ensure credentials are correct. This is priority number two because the number one priority is ensuring the database works as intended. The information will stay largely the same, but it may change in priority later.** |
| **STIMULUS / RESPONSE SEQUENCES** | **Input must be gathered from the user. The arguments the function is checking against is the username and password from the login.csv file that contains the login information for all users. The username and password must match what is saved in the login.csv file, otherwise it will not allow access.** |
| **FUNCTIONAL REQUIREMENTS** | **Two variables are needed: username and password. The function userLogin() will check the username and password to see if they match what is inside the login.csv file. If true, the database will load automatically. The default setting is false, which means the login information will need to be input each time the user wants to use the database initially.** |

## SYSTEM FEATURE 2

|  |  |
| --- | --- |
| **DESCRIPTION AND PRIORITY** | **The function readBooks() will read the books.csv file and its contents, which will then be read into the program. This is priority number one, as without the database, we have no program.** |
| **STIMULUS / RESPONSE SEQUENCES** | **The readBooks() function is automatically encountered upon successful login. The readBooks() function takes the contents of the database and assigns each category of information its own unique index.** |
| **FUNCTIONAL REQUIREMENTS** | **The readBooks() function requires the books.csv file to connect the database information to the user interface which allows the user to search for books. It is also necessary to index each category by assigning the contents to a vector.** |

## SYSTEM FEATURE 3

|  |  |
| --- | --- |
| **DESCRIPTION AND PRIORITY** | **The searchBooks() function allows the user to search for the book they need. This program allows the user to search for any occurrence of one book at one time, providing information back to the user based on their input. The priority of this feature is tied for second, as it is important to not only have the database information at hand, but also a way to search the information and output information back to the user.** |
| **STIMULUS / RESPONSE SEQUENCES** | **Once the login attempt is successful and the database has been loaded in, the UI will prompt the user to search for the book they are looking for. The user will search for a book based on what they know about the book (title, year, publisher, etc.).** |
| **FUNCTIONAL REQUIREMENTS** | **The readBooks() function reads the books.csv contents into the vector for output. Each category will have its own output line.** |

# REQUIREMENTS OF EXTERNAL INTERFACE

## USER INTERFACES

Describe product / user interface characteristics, including standards, style guides, constraints, functionality, and sample screens if applicable.

**User interface characteristics include: the prompts for logging in, searching for books, and asking the user if they would like to search for another book.**

# APPENDICES

## APPENDIX A: GLOSSARY OF TERMS

Define all terms and unique acronyms employed throughout document and specific to project.

**Int main() – The main function controls the flow of the program with most input prompts and directions**

**readBooks() – Function that reads in the books.csv file contents for use in the program**

**searchBooks() – Function that searches the database for user input book title**

**string username – username matching username from login.csv**

**string password – password matching password from login.csv**

**string tryAgain – Variable used to determine what the user would like to do (login unsuccessful – try again or close program)**

**AssignmentTwo.cpp – Calls the functions that read and search the database for user information**

**Login.h – Houses the userLogin() function that determines whether the user has appropriate credentials or not**

**Books.h – Houses the readBooks() and searchBooks() functions that read and output data for the user**

**userLogin() – Function that determines whether the user has correct credentials matching the login.csv file**

**string search – User input for determining if the book if listed in the database**

**string choice – User input character for if the user would like to search for another book or not**

**class Book – Class that houses the data members to be accessed publicly (ISBN, publisher, title, author, year, description)**

**vector<Book> books – Vector that takes in the categories and outputs the information via cout on the user’s screen**

## APPENDIX B: ANALYSIS DOCUMENTATION

List file / document names / provided links to all diagrams, models, additional findings pertinent to technical specification development.

**AssignmentTwo.cpp**

**Login.h**

**Books.h**

## APPENDIX C: ISSUES

List all unresolved issues, TBDs, pending decisions, findings required, conflicts, etc.

| ISSUES | | |
| --- | --- | --- |
| ID | DESCRIPTION | PARTY RESPONSIBLE |
| 1 | **As of this assignment, the program does not deal with the user potentially not already having login credentials. This will be addressed and implemented in the following weeks.** | **Christian McQueen** |
| 2 | **As of this assignment, the program does not deal with the possibility of there being no match or matches for books inside the database. This will be addressed and implemented in the following weeks.** | **Skyler Metzger** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |