Book Repository Project

Software Design for Gaming

A00218352 – Colm Witts

Contents

[Overview: 2](#_Toc467871009)

[Book Functionality: 3](#_Toc467871010)

[General: 3](#_Toc467871011)

[Admin: 3](#_Toc467871012)

[Registered User: 3](#_Toc467871013)

[Guest: 3](#_Toc467871014)

[User Functionality: 4](#_Toc467871015)

[General: 4](#_Toc467871016)

[Admin: 4](#_Toc467871017)

[Registered User: 4](#_Toc467871018)

[Guest: 4](#_Toc467871019)

[System Architecture: 5](#_Toc467871020)

# Overview:

The project is about designing a book management system for a small library. The system is intended to be single terminal only.

The majority of the functionality is related to books available in the library, but also has some functionality related to managing users. Functionality is limited based on the user type (admin, registered user, or guest). A custom doubly linked list class is used in this project, with children for books and users respectively. Data is read in and written out to text files for persistant storage. The UI is a basic, text-only interaface.

# Book Functionality:

## General:

Books are stored a title, an author, an ISBN, and an availability (a boolean value). This information is displayed to the user when they search for books.

## Admin:

Admins can add a new book to the library, or remove an existing book. When removing a book, they can specify either a title to remove, or an ISBN to remove. They can also set books to be available or unavailable, again either by title or ISBN. Finding a book by it’s title is case sensitive.

## Registered User:

A registered user can perform a search of the books available. They can search by title, ISBN or author. When searching, partial matches will be displayer as well as exact matches. It is also **NOT** case sensitive. For example, searching “h” by title would result in every book with a “h” or a “H” in the title.

The users can also check the availability of books, check out books and return books. Similar to how an admin finds books, finding books for these purposes requires an exact title or ISBN. Checking out a book will make it unavailable, and returning a checked out book will make it available once more.

## Guest:

A guest has similar but less functionality than a registered user. They can search for books in the library identically to how a registered user can, but CANNOT check out or return books.

# User Functionality:

## General:

When the system starts, the user will be taken to a user select screen, where they must select either admin, registered user, or guest. Registered users are stored in a linked list of users, and store a username and password used for log-in purposes. This list (including usernames and passwords) is written out to a text file to be stored persistantly. User information is encrypted with a simple Caesar Cipher. Admin, however, is not password protected. Practically, this system lacks any kind of meaningful security.

## Admin:

Admins can add new users to the registered user list. They can also remove users (finding them by user name).

## Registered User:

Registered users are password protected, and must enter their username and password to login. Passwords are case-sensitive, only allow alphabetic characters, and are encrypted once entered.

## Guest:

Guest has no meaningful user functionality.

# System Architecture:

