Project Research Document

Library Management System

X00179223 – Raymond McCarthy

Overview

For my fourth-year project I have decided to build a Library Management System web application. This system will have two main user types, internal library administration staff and library members(customers).

Use cases for the administration staff will include the following actions:
- Add books to the library database.
- Remove books from the library database.
- Edit books in the library database.
- Reserve books for check-out.
- Remove reservations.
- Check out books.
- Check in books.
- Create and remove accounts.
- Log in/log out.
- Edit account details.
- Extend book checkout lease length.

Use cases for the library members(customers) will include the following actions:
- Create an account.
- Log in/log out.
- Search through the book catalogue.
- Reserve a book.
- Check out a book.
- Check in a book.
- Extend book checkout lease length.
- Pay fine for overdue book returns.

USER

A user can create an account and become a library member, then they will be able to search through the catalogue of books available by author, title, category etc. They can select a book and reserve it for a particular date, they can then checkout out or check in a book within those dates.

There will be constraints involved which declare maximum number of days a book can be checked out from the library as well as a maximum number of books checked out at any one time by a library member.

If a member would like to extend their lease of a book they may do so if it has not already been reserved by another user which would cause a conflict.

If a member keeps the book longer than agreed upon in their reservation, they will receive an automated email and incur a fine which can be paid on the website with a third-party payment API.

LIBRARY ADMINISTRATOR
A library administrator can add books to the catalogue as well as edit and remove books. They can create and remove accounts and perform operations for members, such as reserve, check in and check out books. They can also create and edit accounts.

SYSTEM
I will use Spring Boot scheduling functionality to run a scheduled job daily to check if books are overdue and if so, send an automated email to members with overdue books and add a fine/balance to be paid to their account. The user will be unable to check out a book until the fine is paid off.

Tools & technologies

For this project I plan to use several technologies to collectively build my full-stack application.

For the front-end I will use:
- React
- JavaScript
- JSX
- Bootstrap

For the back end I will use:
- Java
- Spring Boot
- Spring JPA
- Stripe (for payments)

For making http requests I will use:
- Axios

Deployment and hosting:
- For hosting my services online, I will use Azure web server
- I will also store my data using an Azure MySQL server

Tools:
- IntelliJ for Java programming on the BE
- Visual studio code for developing the FE

Similar Systems

I originally decided to develop this application because I was asked how I would design a library management system as an interview 'Systems Design' question for my third-year internship placement. I worked through my proposed solution with the interviewer and found the system quite interesting. Due to the fact I was asked this question I was led to assume that this is a relevant and tricky system to implement and for those reasons feel as though it could help me to gain valuable skills and knowledge sought after in todays job market.

In my research I came across a web application called 'libib'. libib is online tool which essentially covers all the aspects of my proposed system and I hope to gain inspiration from it as I dive deeper into this project.

Risk

The greatest risk in this project is my lack of familiarity with the tech stack involved. This project will be my first time using many of the frameworks and libraries listed in this document. The fact that these frameworks are built for use with languages I am familiar with however, gives me confidence that I can pick up these skills relatively fast and use the frameworks built in functionality to my advantage for quick development.