# Library Books Catalog

### Library Books Catalog is a web app created using Java and Angular. This app is developed to allow Librarians the ability properly organize a library. It has two types of users – Guests and Librarians. In order to use the website, you have to register on the website. Regular guests do not have the ability to register as an Admin/Librarian. For a Librarian to be registered, an existing Librarian has to register them through the Librarian Dashboard.

#### Regular users can view a list of all books and are able to freely sort through them to get their desired book. Once the appropriate book is found, the user is able to then view the quantity of the book and decide if he/she wants to place the book on hold.

#### Librarians have the ability to add new books, increase the quantity of a book, or view a list of all books.

# User Stories

## View All Books

|  |  |  |
| --- | --- | --- |
| **User Story ID** | **Priority** | **Status** |
| US01 | This is very critical to the application | INCOMPLETE |
| As a librarian and guest, I need to be able to view all books currently in the library. | | |
| **Acceptance Criteria** | | |
| * Any user type should be provided with access to this feature. * The books should be in pagination format by default. * Book details should include * Display error messages if no books available. | | |

## Admin: View All Genres

|  |  |  |
| --- | --- | --- |
| **User Story ID** | **Priority** | **Status** |
| US02 | This is very critical to the application | INCOMPLETE |
| As both a guest and a librarian, I want to be able to view all the genres so that I can modify and delete them as needed. | | |
| **Acceptance Criteria** | | |
| * Only a librarian should be provided with delete/add features * The genres should be displayed tabular format. * If the number of genres exceeds 10 then it should be displayed in pagination format. * Details include the name of the genres. * Display error messages upon in case of any error. | | |

PROCESS

1. Establish connection to MS SQL using Windows Authentication [DONE]
2. Create first user story[DONE]
3. Create github, filter out unnecessary files, and push first set of files. [DONE]
4. Create book model, dao, service, and api. [DONE]
5. Test api link on postman. [DONE]
6. Create genre model, dao, service, and api. [DONE]
7. After successful tests, setup angular. [6/04/2020]
   1. Divide structure of angular files based on Librarian/Guest relationship.
8. Proper angular routing [6/04/2020]
9. Implement book/genre view in angular web page.
   1. Initially the plan is to simply present the book/genre data on a plain webpage in angular. After confirming everything is functional, we can begin the implementation steps described below.
   2. There should be two different scenarios implemented
      1. The librarian user group has their own viewable section where they can add/delete books/genre.
      2. The other section one that is viewable by both a guest and a librarian. A guest would be able to view a list of all the books and sort by available genres.
   3. To accomplish the points mentioned above, we need to setup a USER table in SQL (name, email, password). Routing and guard should be properly configured in the login page to prevent unwanted access.
   4. Furthermore, it should be possible for a guest to register as a regular user. [ can be implemented later]