Book Library System

This document is aimed to provide requirements of online book library system. Via this system, our stuffs can see these books and borrow them for each working day.

Below are the details:

1. Website

* User login page

After the account is activated, user inputs username and password to login into the system, check if the username and password what user enter is correct. Turn to book list page by default if user’s username and password is correct, otherwise keep in login page.

* Book list page

User can search books via name, tag, description, category, author, support fuzzy search.

User selects a book, system show book detail info. If the book has not been borrowed yet, user can borrow the book, he needs enter return back date. If the book has been borrowed, user can’t borrow the book.

* Borrowed book list page

User can view the books what he borrowed, and he can return back the books what had not been returned back.

- Notification center

User can see notification list what system has notified him. User can remove notifications he want to seee anymore.

1. Back-office

* Log in Page

Pre-set admin/admin into the system.

* User management

Admin can create user for website

In view user list, admin clicks the [Reset] button to re-set password for each user

In view user list, admin clicks the [Active/Inactive] button to manage the user’s status.

* Book category page

Admin can see category list in this page. This page should also support search, delete, edit, create category.

* Book tag page

Admin can see tag list in this page. This page should also support search, delete, edit, create tag.

* Book author page

Admin can see book author list in this page. This page should also support search, delete, edit, create book author.

* Book page

Admin can see book list in this page.

Admin clicks the [Create] button, inputs the book info and save it.

Admin clicks the [View More] button, view the book info and can edit when clicks [Edit] button.

Admin clicks the [Borrowed History] button, show the borrowed history info of this book.

Every book should contain category, tag, author.

Here are our claims:

1. Divide the suitable services for the system.
2. The books and users should store in MySQL.
3. The borrowed records should store in MongoDB.
4. Do not need to create pages, just use postman to mock.
5. The system should have a scheduled job to notify the user whose reservations will expire in 1 day. (send a message using Kafka)
6. Create a notification-service to subscribe the notification and mock sending email to notify the user who subscribed the book.