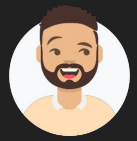
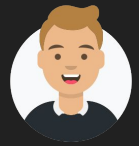
Bookbase

Your library in the digital age.

Skyler Hogan
@skylerhogan



Scott McDonald
@scottmcdonald1



Raleigh Moore
@raleigh-ct

Ryan Mossor
@ryanmossor

Description

We know just how hard it is to find time to read these days.

It's even more difficult to keep track of books you've read and those you want to read in the future. At Bookbase we want to help you rediscover the joy of reading by making your library easier to manage than ever before.

Bookbase is a data-driven, personal library management application. Through the use of "shelves" it allows users to keep track of books based on a number of criteria, including rating, completion status, date-added, last-viewed, and more. At its core, it’s a true book club companion.

Features

- User profile generation and authentication with Spring Security and email confirmation.
- Registered users can add, edit, and remove books from their personal library.
- The bookshelf is dynamically arranged and sorted based on a number of criteria e.g. ISBN, author, genre, date-added, status(read, reading, to-read), etc.
- Users can add and edit personal reviews of books and add ratings if they like.
- Memorable quotes and ideas can also be added to each book.

Planning - User Stories

- As an [unregistered user] I want to [get an idea of the app's functionality and uses] so that [I know why I might register].
- As a [user] I want to [register and login to the site] so that [I can manage my personal library].
- As a [user] I want to [see examples of books I can add to my library on the landing page] so that [I have a better idea of what the site is about].
- As a [user] I want to [manage and add books to my personal library] so that [I can keep track of what books I have read, those I am currently reading, and those that I want to read].
- As a [user] I want to [be able to search] so that [I can find books related to my certain parameters and decide if I want to add them to my bookshelf].
- As a [user] I want to [add a book via Google Books API] so that [I don't have to manually fill out the information].

Planning - DB and Models

User

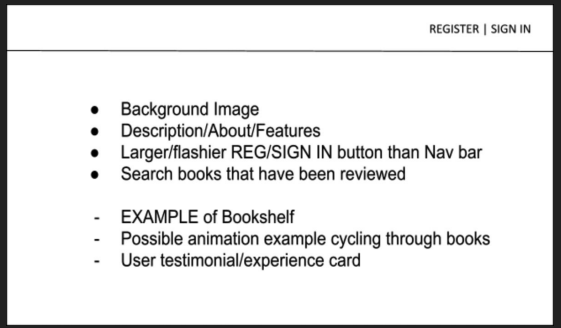
- Name
- Email
- Password hash

Book

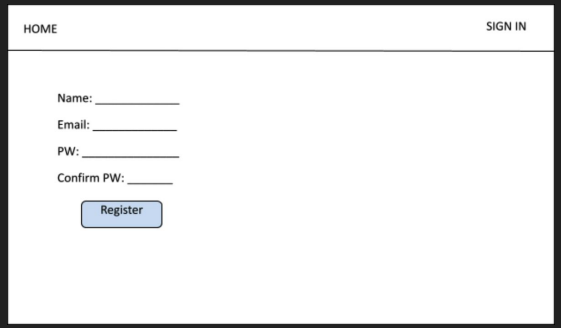
- Id
- Title
- Author
- Pages
- Year
- Rating
- Genre
- Status (currently reading, to read, complete,)
- Day added
- Last Viewed
- Quotes
- Summary

Planning - Wireframes

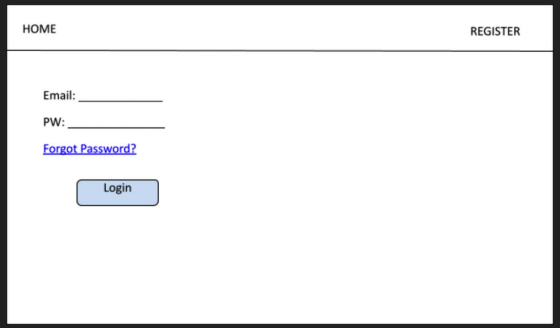
Landing - “/”



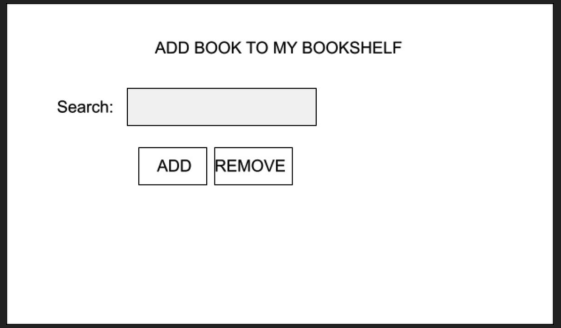
Register - “/register”



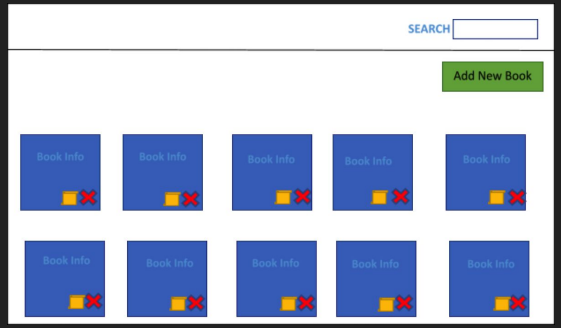
Login - “/login”



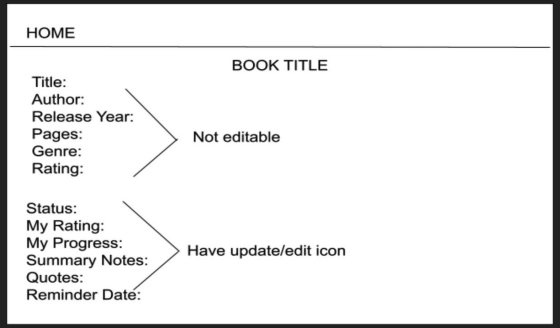
Add - “/dash/addBook”



Bookshelf - “/dash/{userId}”

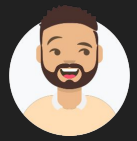


view - “/dash/view/{bookId}”



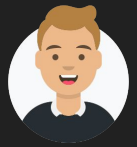
Planning - Divide and Conquer

Skyler



- Managed Trello board while collaboratively designing and delivering new user-stories and features through weekly stand-up meetings.
- Responsible for building out Book object and its instantiation via relational data-binding with an MVC design pattern.

Scott



- Integrated the Google Books API to allow users to search for books.
- Passed JSON information from Google Books API into Book object to be stored in user's bookshelf.
- Implemented a book preview where applicable in search pages.

Raleigh

- Set up user authentication with Spring Security.
- Linked Book object with User object via JPA & Hibernate features.
- Built email confirmation tool and linked with gmail account.

Ryan

- Handled design, layout, and functionality for the homepage.
- Created a dynamic bestsellers carousel and bestsellers list to allow users to discover new titles.
- Implemented a dynamic, labeled view for the current user's shelf page, as well as the ability for the user to sort/filter their shelves.

Technology Stack

- Java 16
- Thymeleaf (Front-end Java framework)
- MySQL (Database for housing book and user data)
- Hibernate
- Spring Framework (Gradle)
- Google Books API (Used to add new books to users' libraries)
- NYT API (For populating dynamic book carousel of current best-sellers)
- HTML/CSS (Building and styling pages)
- Bootstrap (UI/UX)

Demo

What We Learned

- Integration of a third-party API into a Spring based application.
- How to write asynchronous functions to populate web-pages with promise-based behavior.
- How to use CSS frameworks like Bootstrap to create a consistent UI/UX feel across all pages.
- Git essentials like branching, merging, and resolving conflicts through contributing to a group project.
- User authentication/authorization aspects including:
    - Spring Security
    - Functionality of Principal object & UUID tokens
    - Automated email integration in registration process

What's Next

- Link our database to AWS MySQL instance instead of local MySQL server instance
- Automatically add a few classic books to a user's bookshelf upon registration
- Browse various NYT Best Sellers Lists by genre