Final Project Report_05-014

Project Name: The Bookaholics

Team Number: #14-05

Team Members:

- Tristan Barnes: **trba3871@colorado.edu**, GitHub: twisty13
- Ben Garmon: **bega7279@colorado.edu**, GitHub: bennettgarmon
- Cooper Thiel: **coth4827@colorado.edu**, GitHub: cothiel
- Yusef Al-Balushi **yual1945@colorado.edu,** GitHub: TheBenchwarmer21
- Suman Upreti: **suup2300@colorado.edu**, GitHub: suman421159
- Oscar: **osgu7204@colorado.edu**, GitHub: TheBenchwarmer21

Table of Contents:

.

Project Description:

The Bookaholics Web Application is an innovative and interactive platform aimed at book lovers and avid readers. It provides a vibrant and comprehensive experience for discovering, reviewing, and discussing a wide array of books. Developed using Node.js, Express, EJS, and PostgreSQL, allowing users to create profiles, manage book collections, write reviews, and participate in book discussions. Its intuitive interface and user-centric design make it a unique destination for those seeking an engaging reading community.

Video: ***https://youtu.be/KBJ-46r49t8***

Project Tracker (Github Project Board):
https://github.com/users/TheBenchwarmer21/projects/1/views/1

Contributions:

- **Oscar**:

  Some of the features I've created was the ability for users to search a book, which then the application uses Google Books API to fetch Google's collection of books. Added the ability for users to add a book to their collection. Created a home page which is a screen after the user has logged in, which explains what type of buttons the navbar has and how to use the website overall. Added most pop-up messages, which some of the examples include messages when a user added to a book, if books are already in collection, user added review, etc.

- **Yusef Al-Balush:**

  Contributed to the structure of directories and the ability for any user to register a certain account and log into our Bookaholic website. The Login and Register also takes into account whether the user entered the correct information and whether the user is trying to register an account using details used on a pre-existing account. Developed my Book page which is the book collection for the user, in which the user has options to either remove the book from the collection, read the book from the collection or add a review to the collection. Added banners, pop-up messages, and stylistic preferences on almost each page of the website to make it look presentable and professional to the user. Contributed to the welcome message, showcasing what the website is about and who we are as a community of bookaholics.

- **Suman Upreti**:

  Contributed to development of backend route reviews to fetch and display all of the book reviews, joining them with corresponding book details. Implemented logic to group reviews by book title and to calculate each book's average rating. Additionally played a role in the frontend development, designing a responsive and user-friendly interface using HTML, CSS, and JavaScript, which included features like expandable sections for additional reviews. Also contributed to moodle for adding new reviews, ensuring error handling, and enhancing the overall user experience.

- **Tristan Barnes**:

  Contributed to the development of core features such as book review posting and managing user profiles. Implemented the docker-compose and ejs files into our repo for all members to use. Created SQLdatabase and implemented the insert files to have sample data to show up each time the site loads. Worked on an otherreviews.ejs file but it got scrapped during development due to another solution being found during development of reviews.ejs

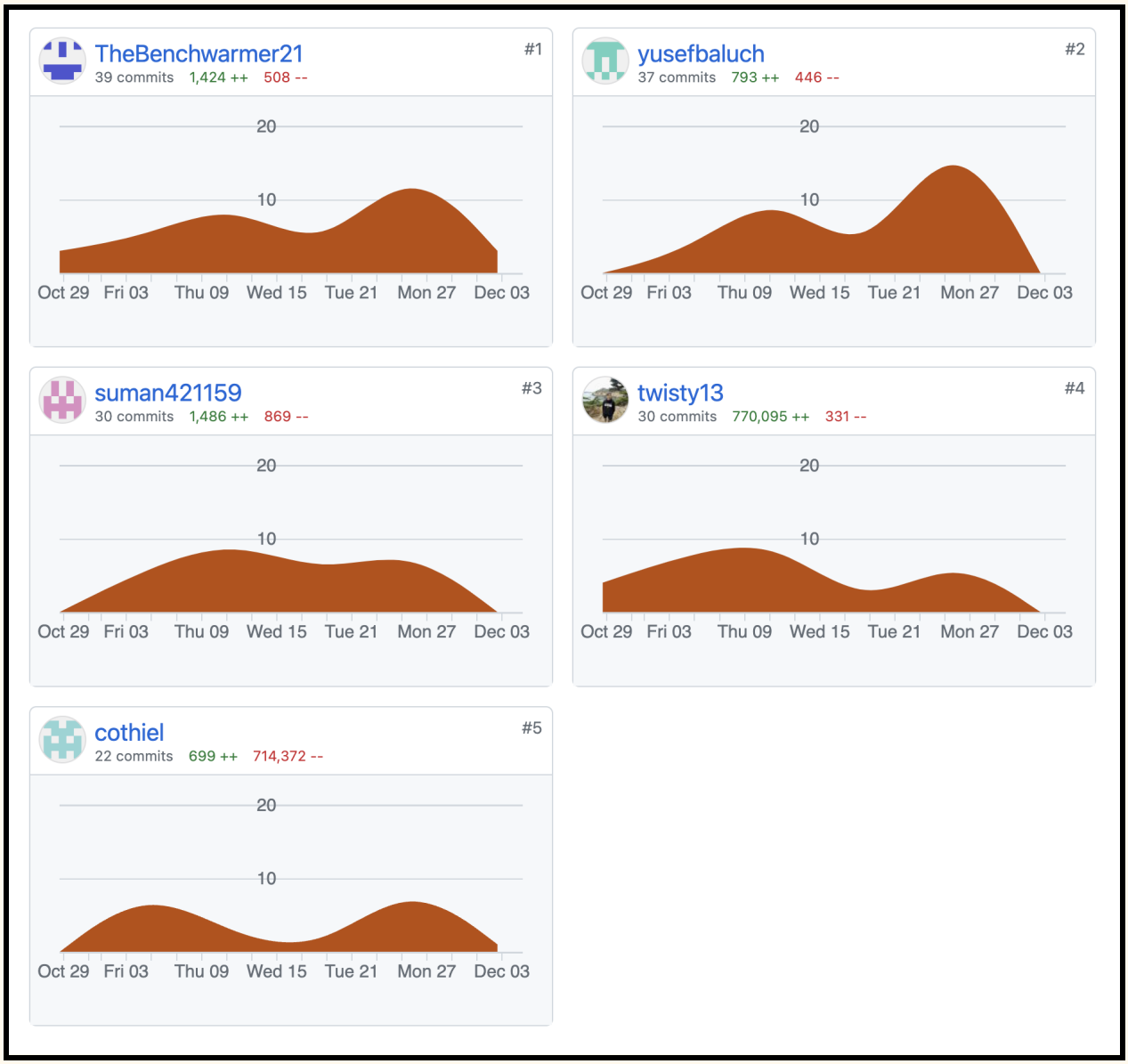
- **Ben Garmon**:

  Worked on the navbar and welcome screen, and deployed the website on azure.

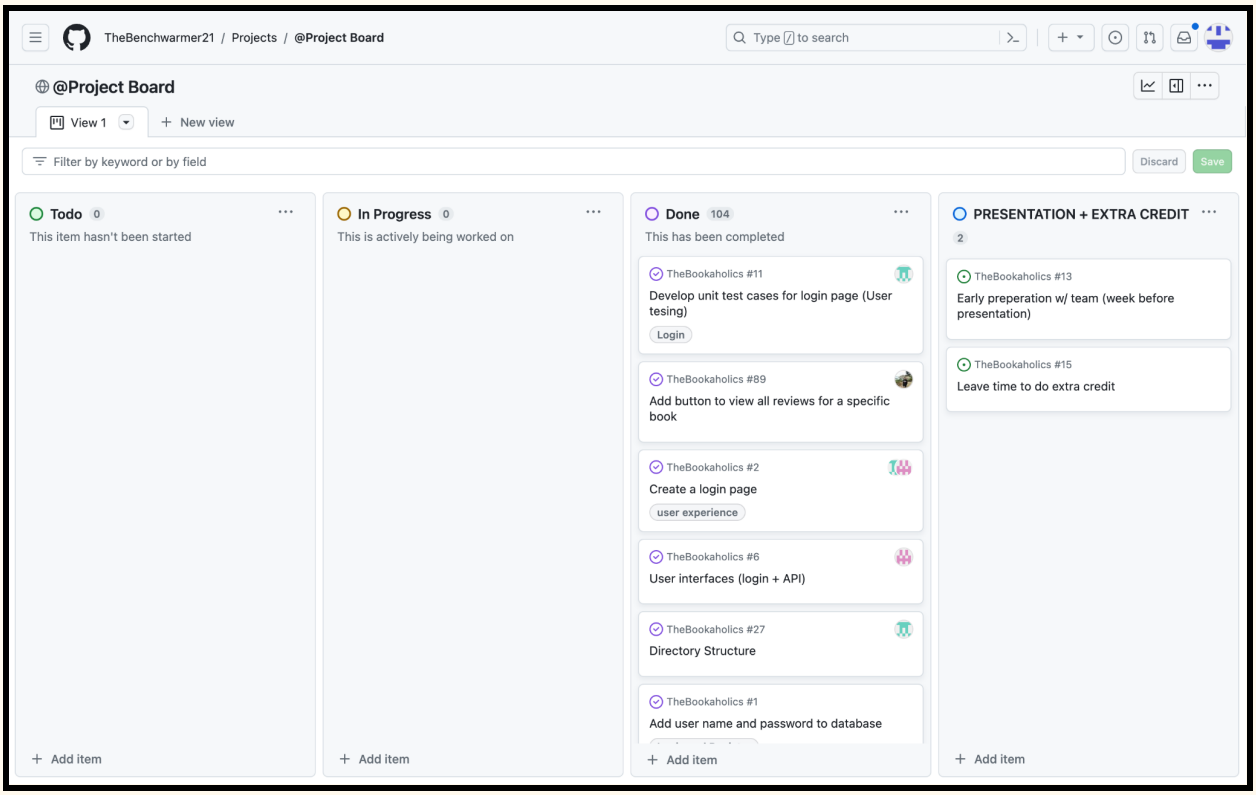
- **Cooper Thiel**:

  Developed myreviews page including review editing/deletion. Supported the development of both other reviews and reviews. Ensured consistent design language and cleanliness throughout the website. Designed a use case diagram, developed unit test cases, and enacted UAT plan. Supported other members when encountering issues/solving bugs.

Commit-based Contributions:

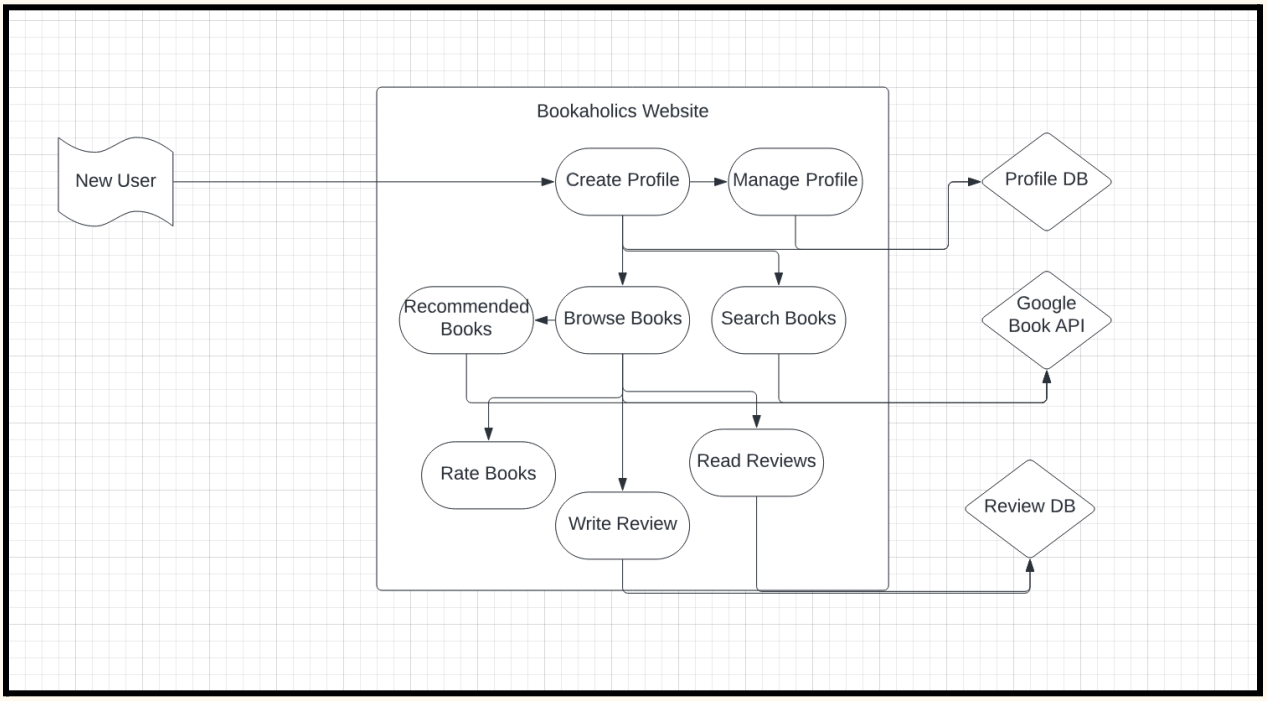


Project Board:

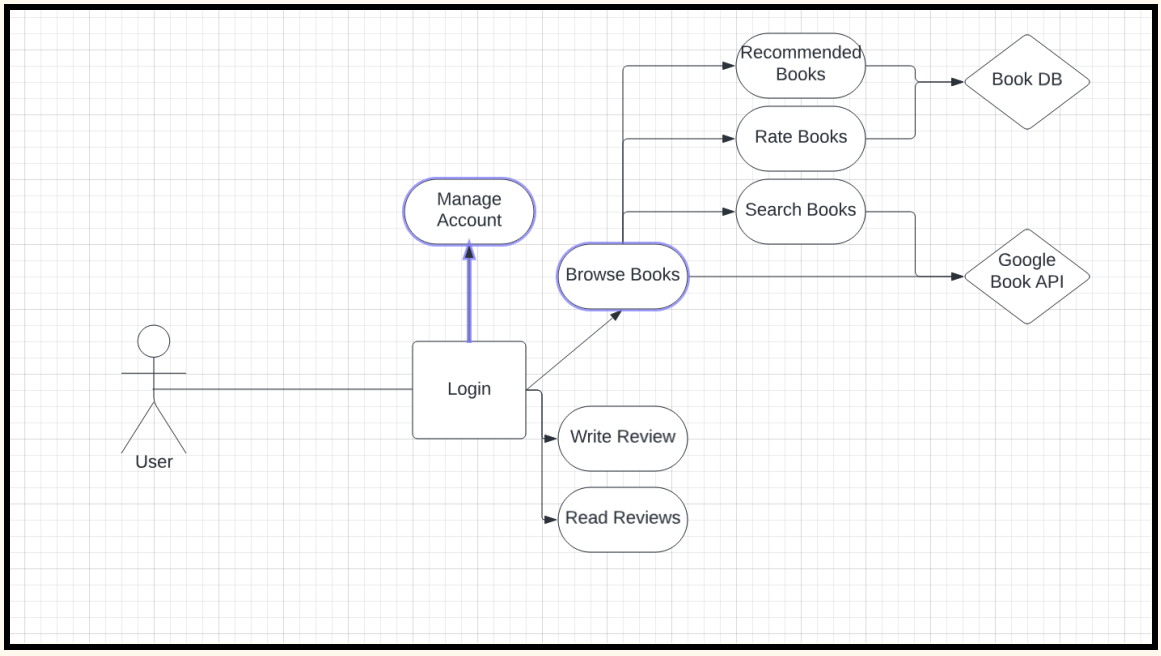


Git Repository Link (VCS): https://github.com/TheBenchwarmer21/TheBookaholics

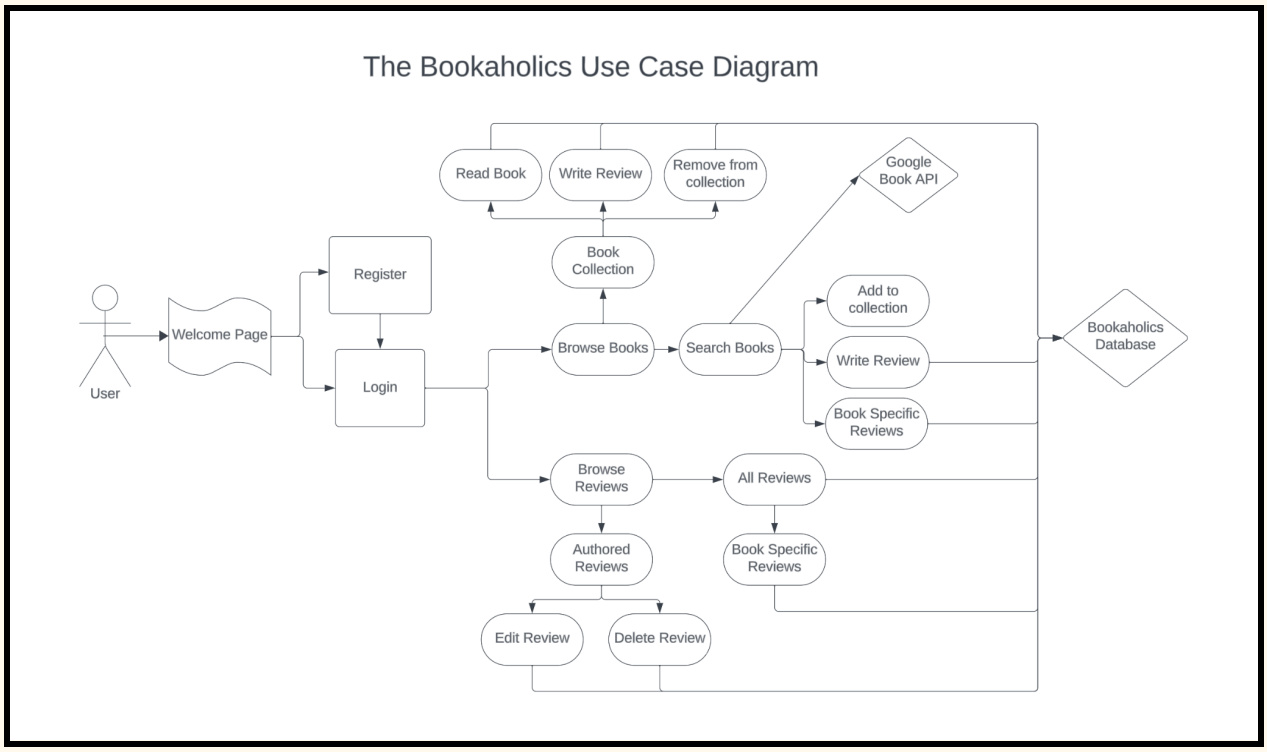
Use Case Diagrams:



Early Development Use Case Diagram 1



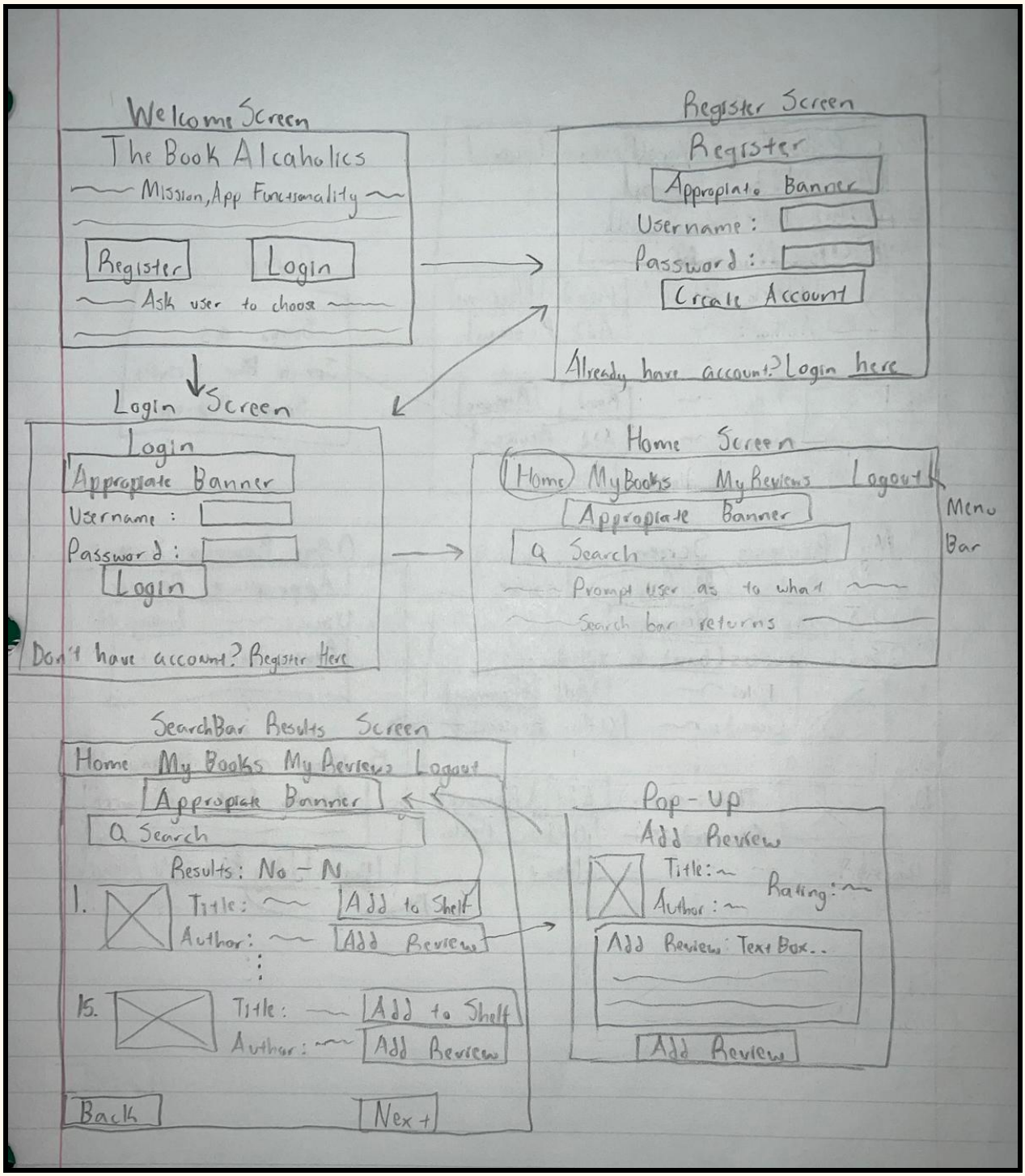
Early Development Use Case Diagram 2

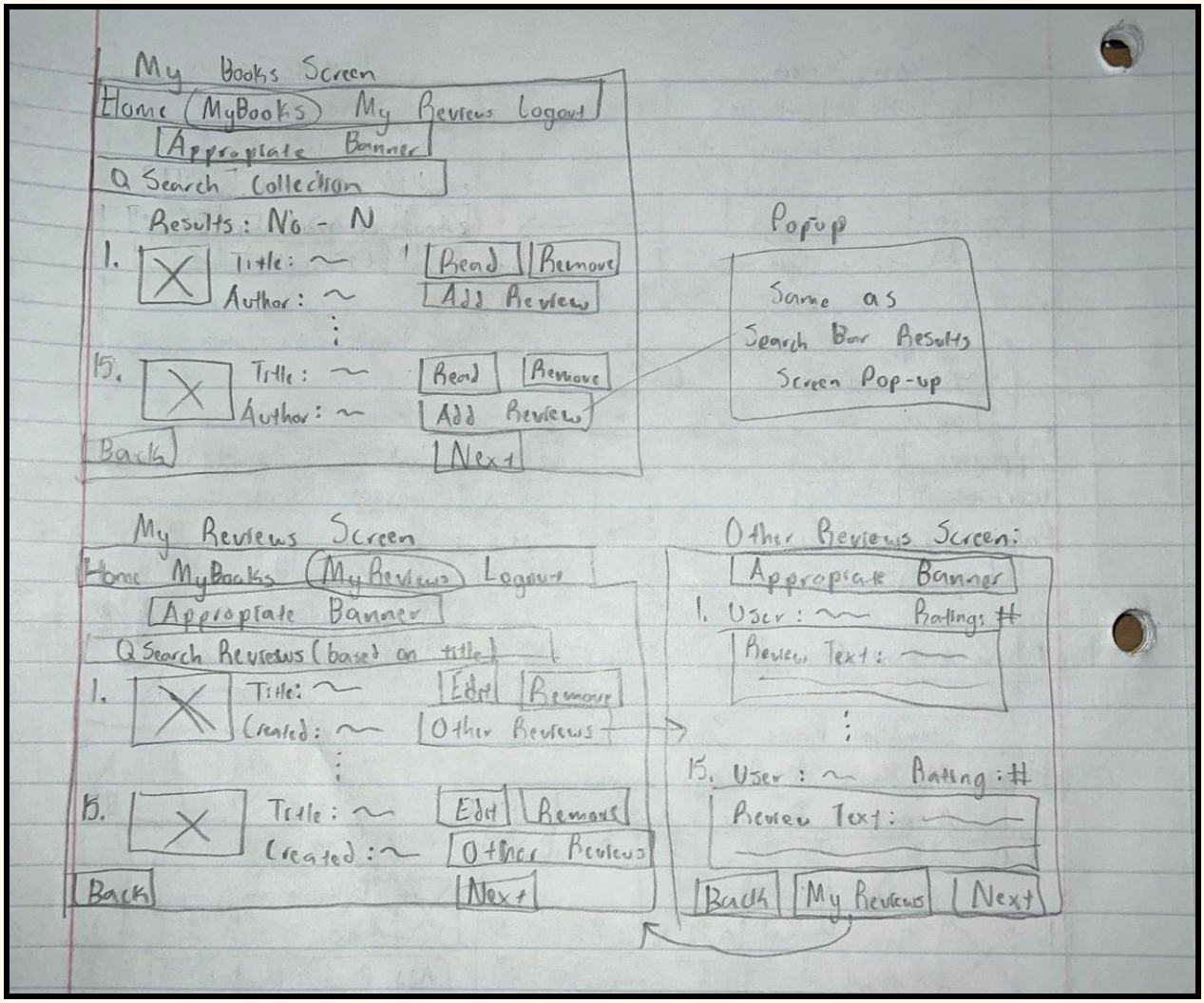


Final Use Case Diagram

Wireframes

Below are our wireframes for our application that displays the user interface of our overall application.




My Books Screen
Home MyBooks My Reviews Logout
Appropriate Banner
Q Search Collection
Results: No. - N
1. Title: ~ Author: ~ Read Remove Add Review
15. Title: ~ Author: ~ Read Remove Add Review
Back Next
Popup
Same as Search Bar Results Screen Pop-up
My Reviews Screen
Home MyBooks MyReviews Logout
Appropriate Banner
Q Search Reviews (based on title)
1. Title: ~ Created: ~ Edit Remove Other Reviews
15. Title: ~ Created: ~ Edit Remove Other Reviews
Back Next
Other Reviews Screen:
Appropriate Banner
1. User: ~ Rating: #
Review Text: ~
15. User: ~ Rating: #
Review Text: ~
Back My Reviews Next

User Acceptance Testing Results

Test: Successful registration
Observations: User successfully registered, user appreciated the redirect to login after a successful registration.

Test: Successful login
Observations: User successfully logged in, user initially made a mistake in their password, but quickly realized their mistake as a result of the on screen “incorrect username or password” message.

Test: Book search
Observations: After logging in, the user went to search for “bible”. Search results correctly displayed various titles containing the keyword “bible”. The User added the book to their collection and subsequently wrote a review for the bible. The user appreciated the partially filled beer images indicating average google book rating.

Test: Write Review
Observations: User began to write a review for the bible, all functionality worked as expected and user did not make any unexpected actions. The User recommended that rating be a dropdown menu rather than the user typing the rating themselves, this feedback is actively being implemented.

Test: Book collection
Observations: After writing a review, the user went to view their book collection, upon viewing the page, the user clicked on the “read” button to read one of their added books. This function successfully redirected the user to the google books page for their book. The user appreciated that the button opens a new tab, rather than simply redirecting the current tab.

Test: Reviews page

Observations: The user went to the reviews page to view all posted reviews. The user added “Dracula” to their collection via a posted test review. The user subsequently left the page.

Overall Observations

The user did not make any unexpected/unaccounted for actions, the general flow of the website seemed to be in good shape. The user did not indicate any confusion regarding any of the website features nor any aspect of website navigation.

Deployment Information

Deployment Environment link:
http://recitation-14-team-05.eastus.cloudapp.azure.com:3000/login
The application can also be run locally, which involves cloning the project to a local machine and running it with Docker. This process is explained more in depth in the README.md in our project repository under “Instructions On How To Run The Application Locally”:
https://github.com/TheBenchwarmer21/TheBookaholics