Web Systems Development (ITWS-2110)

Final Report:

BookWorms

Group 5

Section 1

Instructors: Thilanka Munasinghe, Satej Sawant

Version 1.0

December 11, 2018

Prepared by

Davin Martinez(berkos@rpi.edu)

George Marinopoulos (hsit@rpi.edu)

Jonathan Byers(manohr@rpi.edu)

Nikita Serebryanik (serebn@rpi.edu)

Table of Contents

1 Introduction

In the last few decades, the cost of attending colleges has increased exponentially. Tuition alone can easily cost more than $50,000, and many other hidden fees and expenses add up to several thousands on top of that. Scholarships and loans are often used to fully or partially cover tuition, however there often still lies the problem of the above mentioned secondary fees. First amongst these secondary expenses, are textbooks, which can easily add several hundreds, if not thousands, of dollars a semester to student debt. Cost savings measures exist, such as renting books for a semester, or buying them used, however this is not often applicable such as when a new book is needed, or costs nearly as much as the buying the book new.

BookWorms was developed as a potential solution to this time and economic drain on students. Taking the same idea as seen in today's "sharing" economy, such as the one that started it all, Uber, and retail sharing giant, Poshmark, BookWorms aims to offer a convenient method by which students can simplify their lives even more. The basic premise of this application is that it allows students on the RPI campus to post, request, and lend textbooks required for classes without the hassle of walking to the library or shoveling money to online sellers. By creating an account, students can choose to update their accounts with specific books they own or simply search for one that they need by typing in the title of the book, the author, or simply the ISBN. The service will then provide contact information for the book's current owner, his/her copy, and allow for the potential borrower to contact them and transfer temporary lease of the book.

2 Project Objectives

2.1 Mission Statement

To provide a convenient service for students to cut down on their textbook costs and reduce the amount of time needed and inconvenience of borrowing books.

2.2 Customer Requirements

The customer base for this application would be RPI students and, potentially, faculty. The requirements for such an application would be user-friendliness and reliability, allowing for a quick search and transaction of items. Not only this, but the app should be intuitive so that users are able to easily navigate to the needed information without scrolling through or accessing unnecessary pages.

2.3 Technical Specifications

- For the front end of this project, HTML, CSS and JS will be used to achieve a more user-friendly experience.
- For the backend side of this project, MySQL will be used as the main database to store identification information about users and all data about the books that are in the system.
- For authentication, a standard login/password combination was used along with data that for books including book title, ISBN, current possessor, and the book's location.
- Each time a transfer of books happens, the user will need to provide a new address and change the location link in the table to correspond to the location of the new user.

3 Existing Technologies

Before discussion of the creation of the team's web application, it was beneficial to do some preliminary research on similar technologies that are currently in existence. Getting a sense of what is already available to students and dissecting such products was helpful in creating an original design that amalgamated that which was found, as well as lacking, in the existing applications.

3.1 BookMooch

It came to no surprise that in today's world of technological innovation, an application such as BookWorms would already be an existence. The first application that came across in research was BookMooch, where its homepage can be seen in Figure 1 below. It allows users to browse, create an account, and to log in to their accounts, with an additional tab for reading up on exactly what the service entails. There is also a points system in existence where users can earn points each time they mail a book out, search for a book, give away a book, along with other means. These points can then be used to request books from others, or to receive books mailed from either within or outside the country.

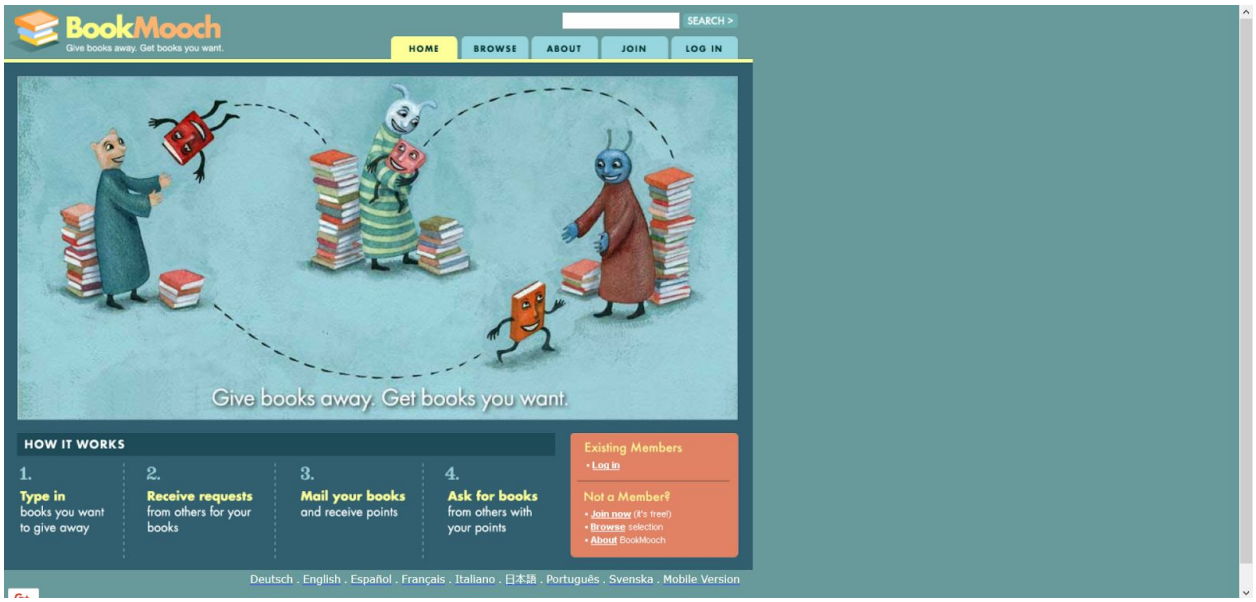

Figure 1: BookMooch Homepage

However, upon further inspection of this website, it can be seen that the front end is lacking in aesthetic appeal (and functionality). The landing page offers an awkward whitespace (or bluespace, if you will) on the right side filled with no information or graphics. Ignoring that, the homepage looks very neat and user friendly, but a deep dive into the other tabs, such as the "Join" page (Figure 2), proves otherwise. Though it has the necessary components to get the basics of creating an account, it does not draw the user in as much as it should. The search functionality is incredibly slow, and seems to not work properly, returning several individual entries of books, instead of one entry with several copies(a feature that is included but seemingly not working). The search function is also inconsistent, and will return vastly different results for searches made seconds apart and identically.

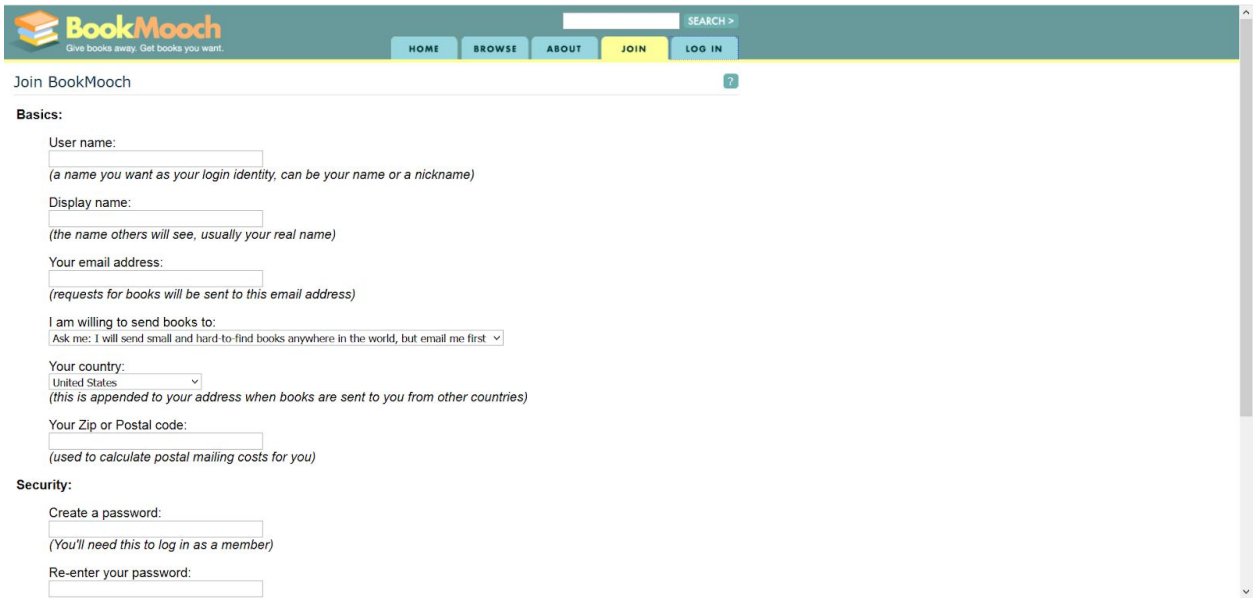

Figure 2: BookMooch Join Page

3.2 Lendle

The second application that was found that closely matched BookWorms was Lendle (Figure 3). As E-readers have become a viable and portable means for bibliophiles to take their beloved literature, an app such as Lendle is an E-reader's dreams come true. This app allows users to share books with others via their devices. Similar to BookMooch, there is a points system, called "borrow requests" that can be earned through engagement with the website, whether it be owning books or lending them to others.

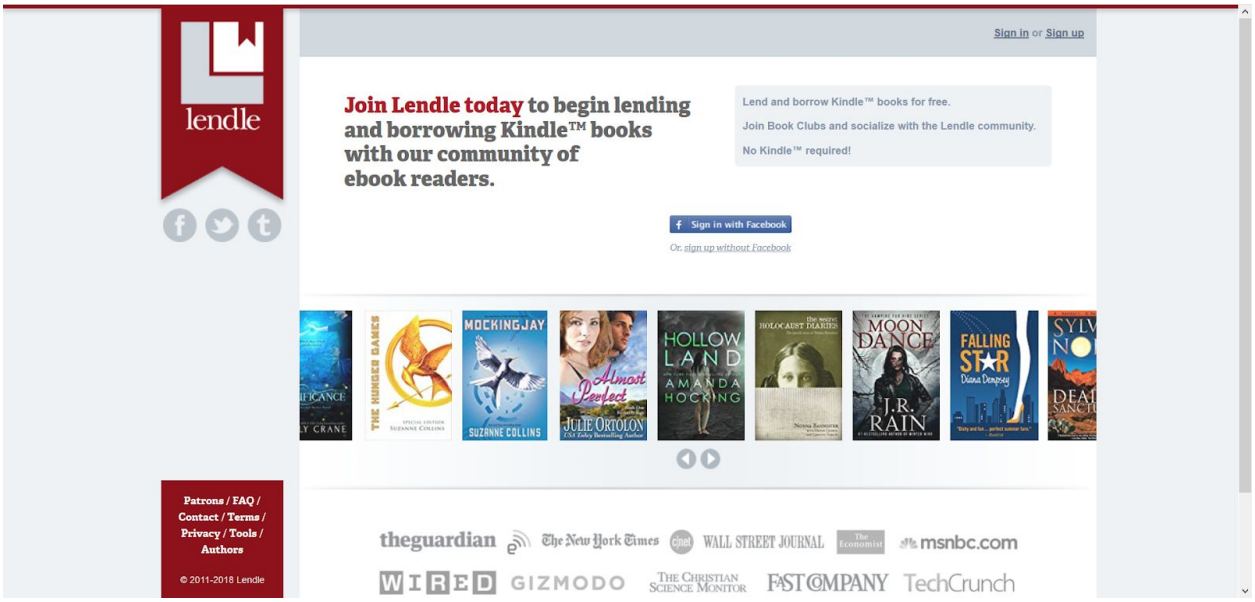

Figure 3: Lendle Homepage

The front end of this application is much more sophisticated and robust than that seen in BookMooch, even with their two respective home pages compared. One of the overarching downsides to this application is the selection of books available. Since it is geared toward E-readers, only E-books are available on the website. Not only this, but such E-books are only Kindle compatible while the service is only available to users located in the United States. This is very inconvenient for those who are interested in such a service but live outside of the country or use some other form of E-reader. While it is robust in terms of front end design, it is less so in regards to back end.

4 Concept Development and Design

4.1 Gantt Chart

Our Gantt Chart is listed below in Figure 4, as of December 3. We completed everything on time, according to our schedule.

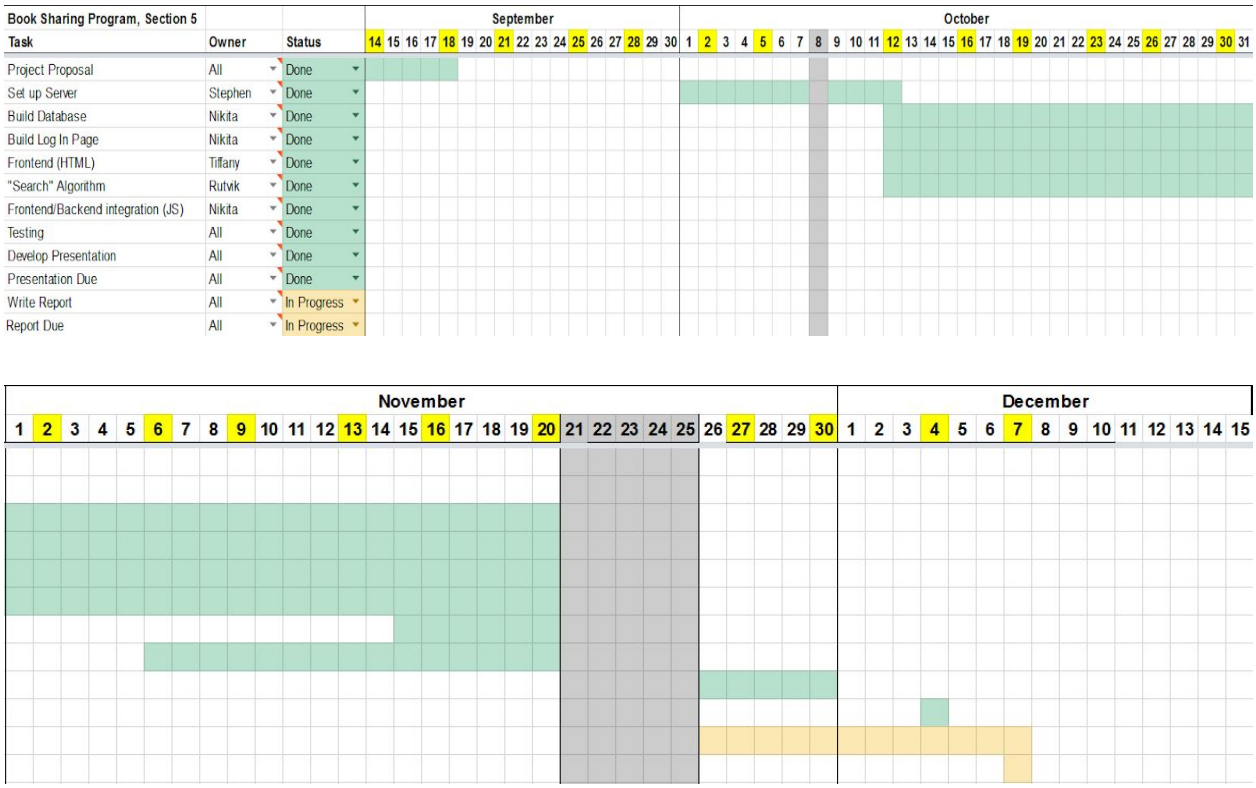

Figure 4: Gantt Chart

4.2 Prototypes

4.2.1 Site Map and Individual Page Wireframes

Figure 5: Site Map

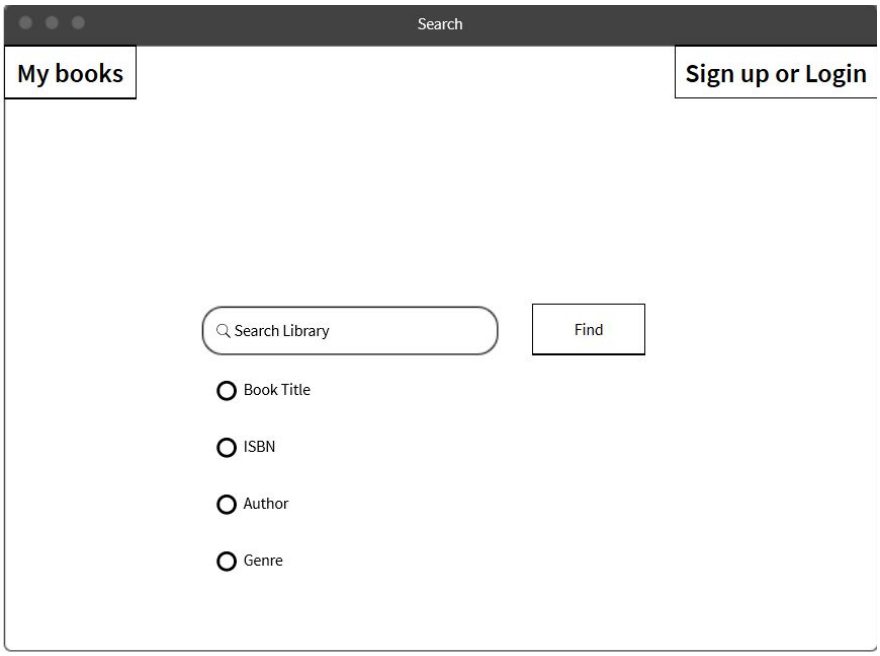

Figure 6: Splash Page

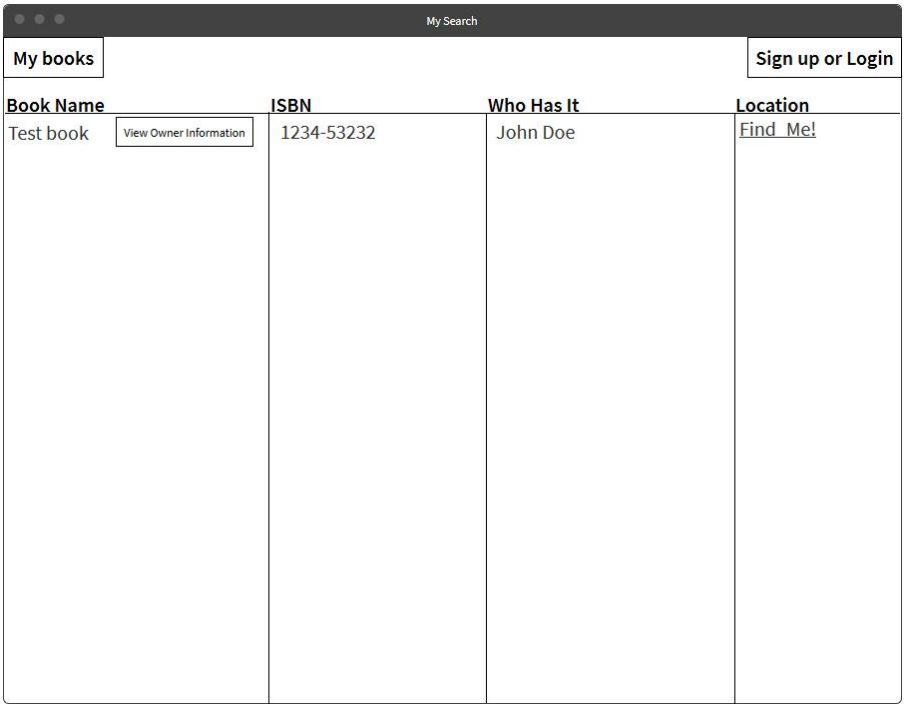

Figure 7: Search Results Page

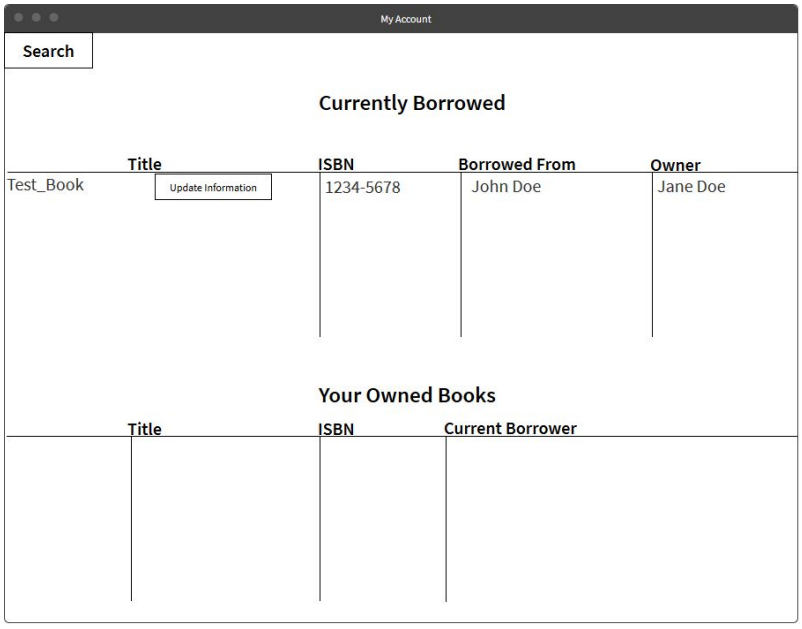

Figure 8: Users Account Page

4.2.2 Final Design

Our website has an intuitive, simple, and clean design. On the main page, the user is able to see all of the functions of our web application. This eliminates the need for users to aimlessly click around the application, saving valuable time to do other tasks. The design of the search page is straightforward without any unnecessary elements. Figures 9-16 display screenshots of each individual page of the web application.

5 Subsystem Analysis and Design

5.1 Subsystem 1: Server Setup

Our website is currently hosted on 000webhosting, a free service that offers web hosting and database support. 000webhosting comes with integrated phpmyadmin, which allowed the team to easily duplicate the XAMPP environment we had been using for development, and effortlessly scale the project. In the future, if the application grow into high demand, it would be very easy to purchase more bandwidth from them and scale the project even further.

5.2 Subsystem 2: Database

Our database is hosted in MariaDB, and consists of two seperate tables, users and books.

The users table contains the ID, which is the auto-incremented primary key, username(unique key), a password hash, and a timestamp for when the account was created. The username is not unique on the database end, however before every user is created, a query is run via the PDO to see if the username already exists.

The books table contains the ID, which is also the auto-incremented primary key of the book, the book title, the ISBN, the book's original owner(foreign key), the current possessor(foreign key), the genre, and the book's current location.

5.3 Subsystem 3: User Interface (HTML/CSS)

The CSS mostly deals with the form controls that are used consistently throughout the website, and various display elements. As seen in Figure 10, which displays the search results, there is a button on the top left hand side to return the user to the search page. On the top right hand side is a button to allow the user to log in to his/her account, or, if an account has not yet been created, to create an account. The design of this page is kept simple enough so that users are not distracted by any overwhelming colors. The purpose is for a streamlined user experience that is intuitive for first-time visitors to the website.

Similarly, in Figure 13, the Users Account page, the two buttons on the top left and right hand sides allow the user to return to the search page and to create an account/login, respectively. On this page, there are two separate tables that display what the user currently owns as well as what the user has currently borrowed from another user.

5.4 Subsystem 4: Search Algorithm

The "search algorithm" is a query of the mySQL database that is constructed via a PDO/PHP and then submitted to the database. Depending on which radial button was pressed (ISBN, Title, genre, author) a different field from the the books table is selected for the query to look through, paired with whatever the user has entered into the search box.

Figure 9: Search Page

I

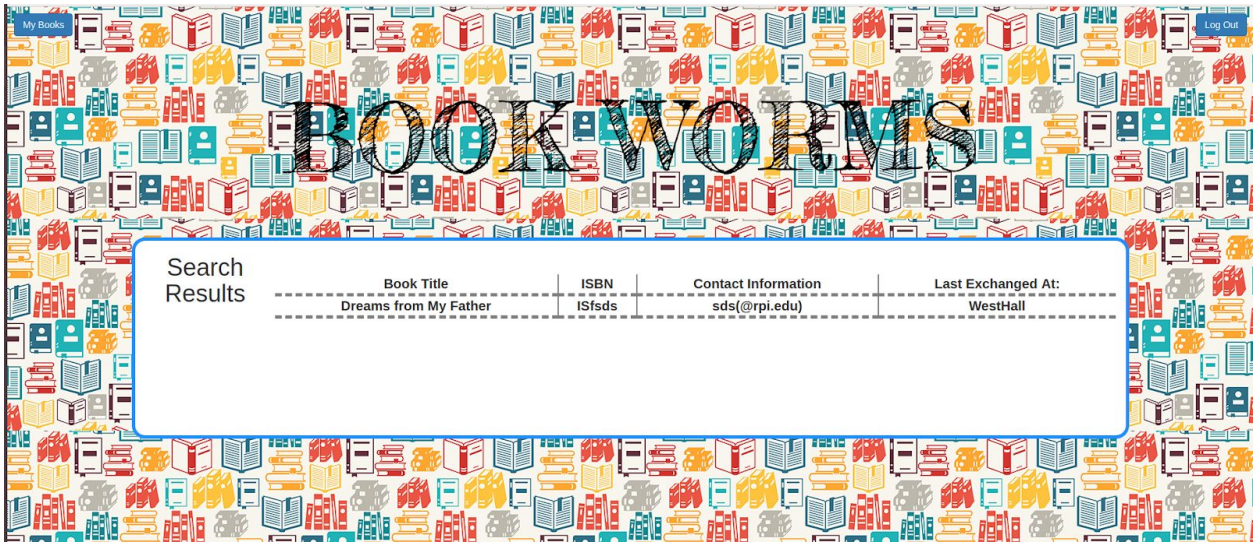

Figure 10: Search Results Page

5.5 Subsystem 5: Transfer Ownership of Books

The transfer book option, on the user account page allowed a user to transfer current possession of the book to someone else simply by entering their username and where the transaction happened.

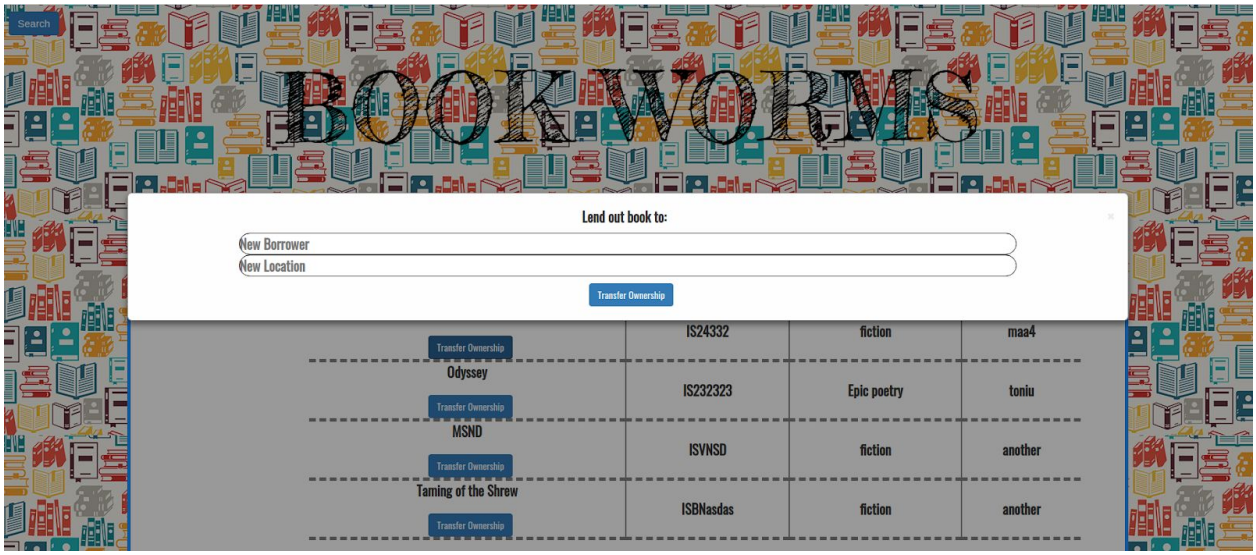

Figure 11: Transfer Ownership Modal

5.6 Subsystem 6: Add Book to Personal Collection

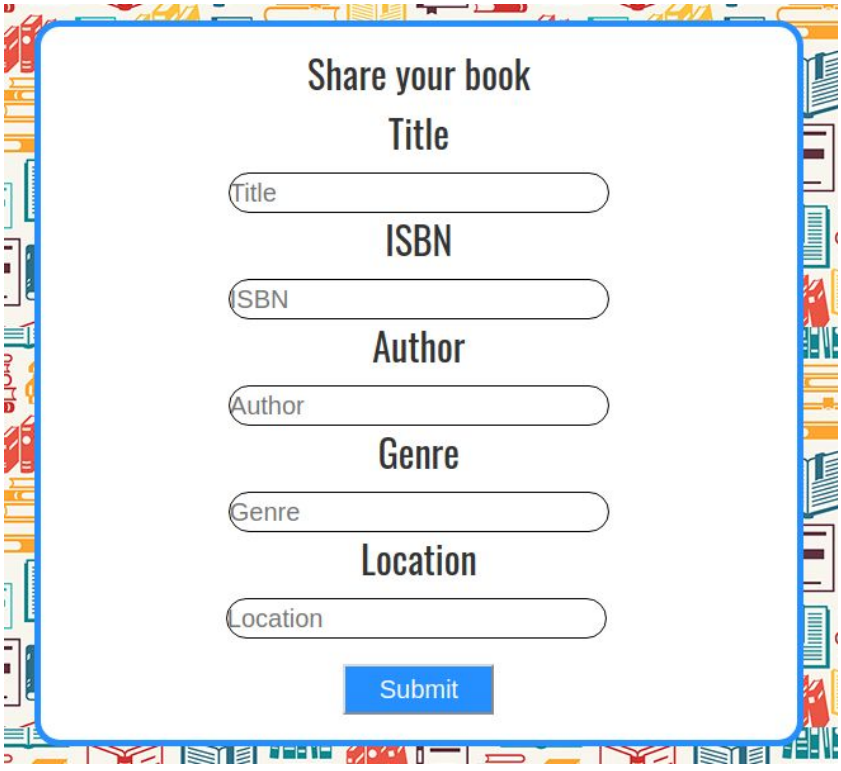

Figure 12: Add Book Modal

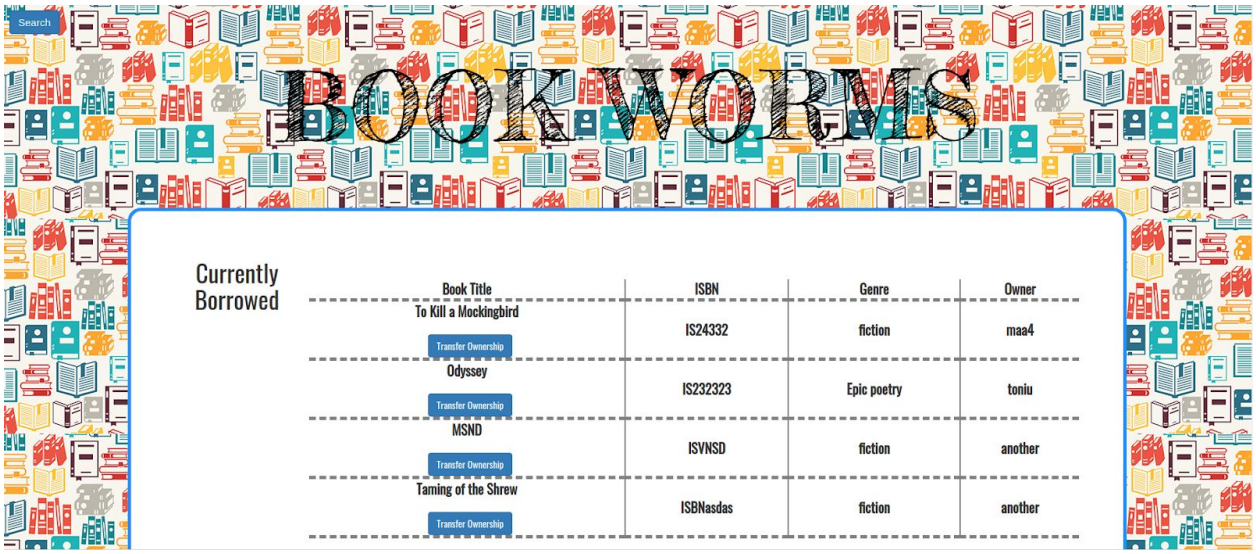

Figure 13: User Accounts Page Currently Borrowed

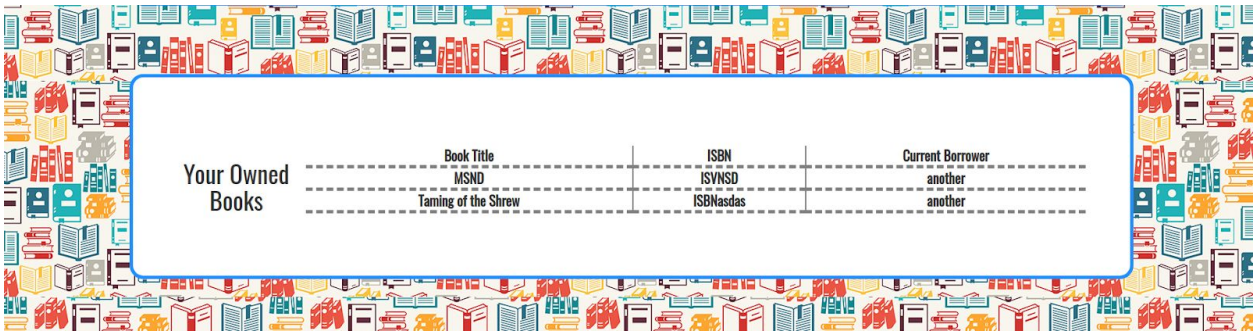

Figure 14: User Accounts Page Your Owned Books

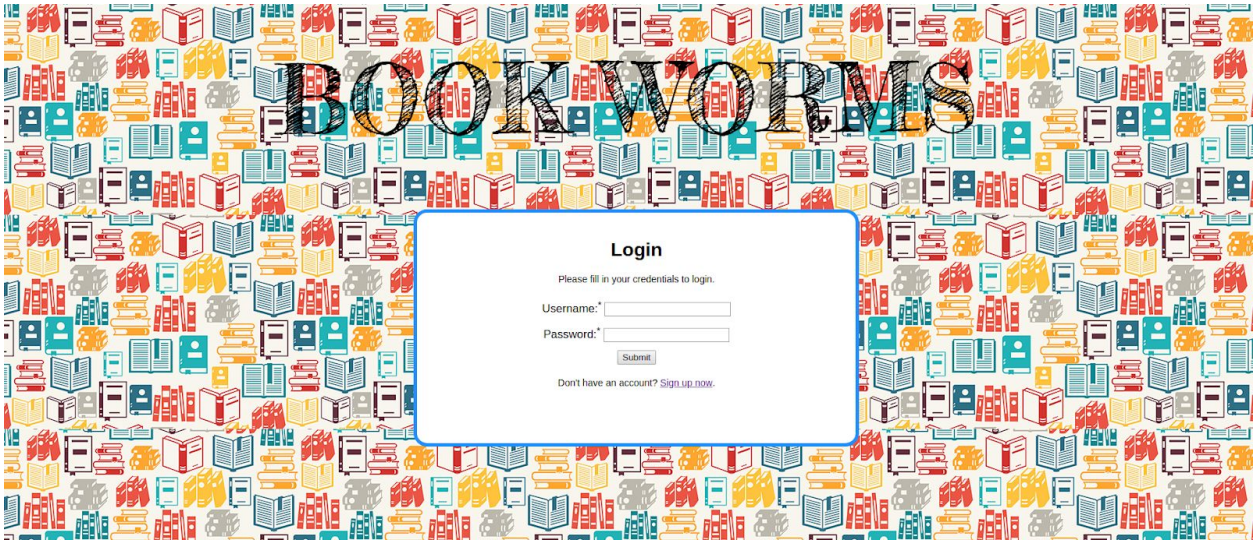

Figure 15: Login Page

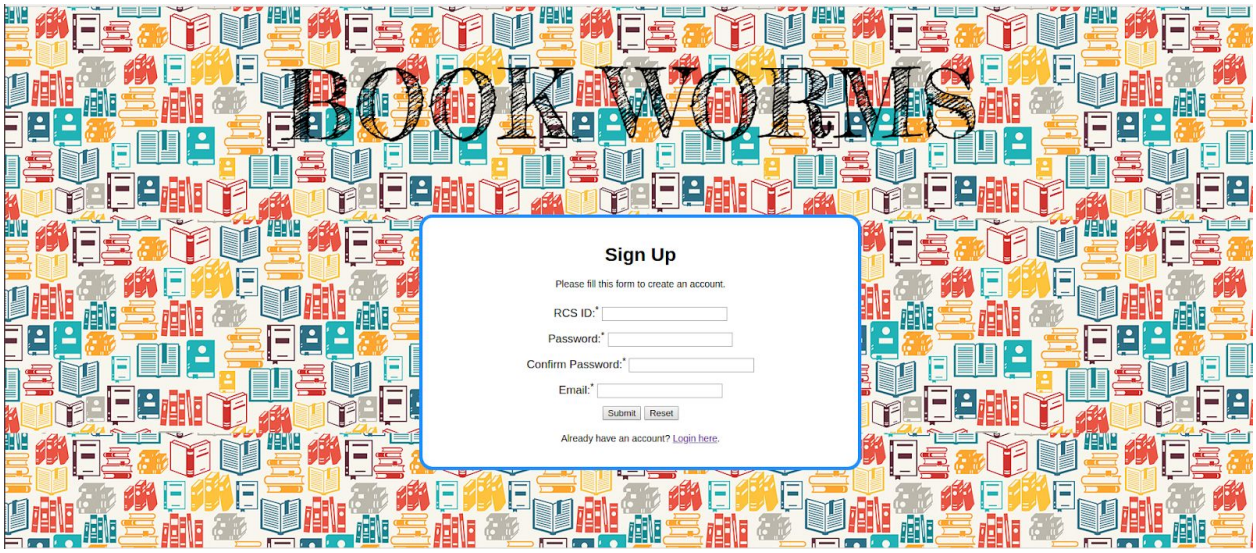

Figure 16: Registration Page

6 Testing and Analysis

6.1 Future Development Considerations

As for future development, there are many possibilities that could be implemented a few that will improve the efficiency and foundation of the web application. One of these developments can be inspired from those technologies already in existence, as explained in Section 3 above, where there is a points system in place to act as motivation for users to interact with the application and post books. As an extension to this system, giveaways or raffles can be used as further motivation to earn points where users can use their points to get a chance to win gift cards, etc. Another addition to the application that can be taken into consideration in the future is to incorporate E-books, such as those available in Lendle (Section 3.2). This would expand the selection currently available with just physical copies of textbooks. Also, if more time was available in developing this application further, the idea of expanding this to outside the RPI campus could be examined. The application could initially apply to the Troy area, then to the Capital Region, and on and on.

6.2 Problems Encountered

One of the issues faced in the development process was hosting. Originally, the plan was to use RPI's hosting service. However, it was difficult to figure out what was going on, and so the fallback was Git. Due to the hosting limitations in Git in the sense that there was no PHP or MySQL support, this was also not a viable option. In the end, the team turned to 000webhosting as it provided more than enough resources(1GB database limit, 10GB bandwidth) for free and

has integrated mySQL support (comes with its own phpmyadmin client!). Minor issues remained though, as security features built into modern web browsers and 000webhosting mandated many minor changes, such as http into https for loading jquery, and changing our original file structure (folders for resources and so on) into a flat file structure(No organization).

The main other issue we faced in the project was using Github. As this was the first major project that most of us worked collaboratively with others on Git with, it was certainly a learning experience. During our development we realized that if people forget to push their current work on git as soon as they were done, others would continue to do work on older versions, which would eventually lead to conflicts. In hindsight, we should have made use of branches earlier, and had a clearer separation of work.

6.3 Lessons Learned

Throughout the project, we learned that this idea could make a difference to people in the RPI community. We also learned that there are many parts to a web application such as ours, so working as a team surely makes developing the application more efficient as well as more exciting. From the beginning of the project to the end, we enjoyed creating this idea and bringing it to a real-life solution and it gave us strong insight on how to be a part of a team striving towards one goal. Not only was teamwork a lesson achieved through this project, but the team was also able to effectively utilize and further develop the concepts gleaned from topics taught since the beginning of this course.

7 Conclusion

In the initial stages of planning this project, the team set out to make a web application using the concepts learned in Web Systems Development. The project sought to help the RPI student body along with faculty to better make use of both time and money by emulating a "sharing" economy of textbooks for courses. With the final product created, the team accomplished all of these goals. The two main areas of focus on which the team concentrated the project were getting real data from a database and using Javascript to the application more user friendly and functional. The team also used various methods to create the overall application, including SQL, JavaScript, and HTML/CSS. Not only were the physical coding concepts intertwined into the project, but those such as the importance of information architecture and project planning were taken into account as well. Information architecture on both the user interface and database ends were used for a straightforward user experience. A Gantt chart, as seen in Section 4.1 above, was used for the project timeline and setting deadlines for all team members. In the end, a prototype that contained the ability to create a user account, lend and search for books was successfully built and concepts acquired throughout the semester were applied in a "real-life" situation.

8 Relevant Links

8.1 GitHub

The team used GitHub as the source of version control. The following is the link to the GitHub repository, where all code files for the project are located, as well as the README file:

Note: Repository was private until 12/11/18.

https://github.com/VNGLR/Websys-project/tree/final_release_branch