SI539 Final Project

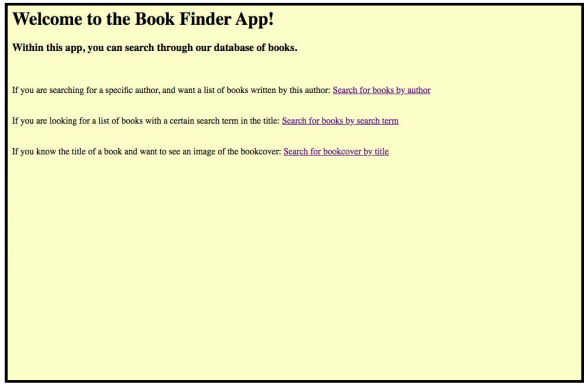
By Sara Gladchun

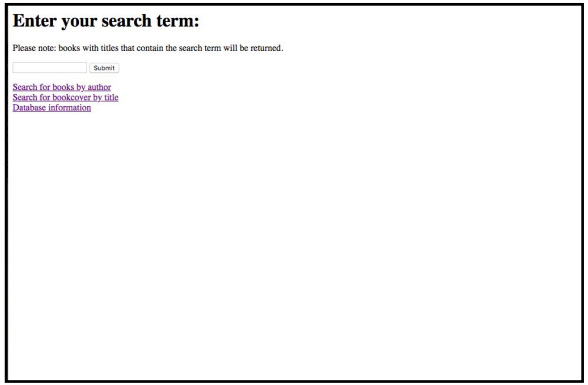
Final Project Overview

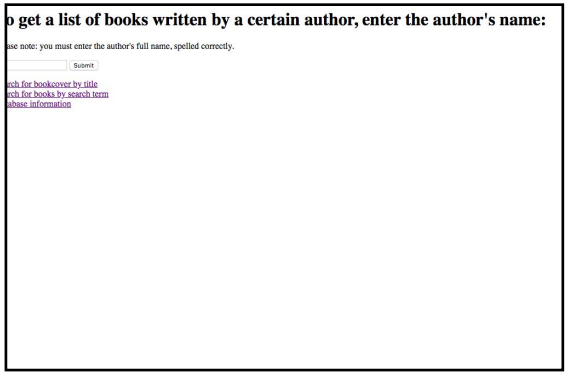
Building out the frontend and UX of a Python flask application previously built (in SI507, winter 2019). My objects for this final project were:

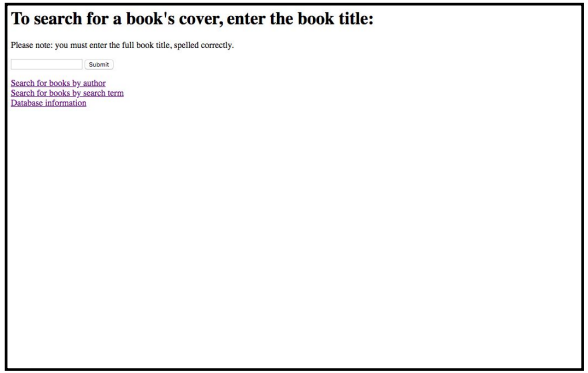
- Build out a portfolio piece to display back and front end skills
- Create a better UX for anyone using my book finder app
- Make my web application more aesthetically pleasing
- Add CSS to this piece (previously no CSS and minimal HTML)

Before









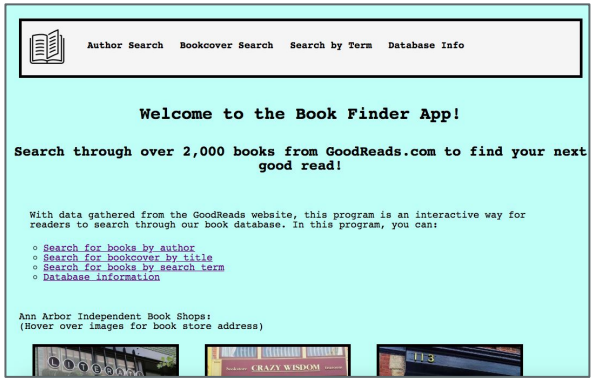
- NO CSS or styling
- Minimal HTML (no divs, classes, IDs, etc)
- Horrible flashing background in dark yellow and light yellow to reach javascript requirement of the assignment
- No unifying color throughout the page (some pages white, some yellow)
- No header, footer, or nav bar

Example of Previous HTML

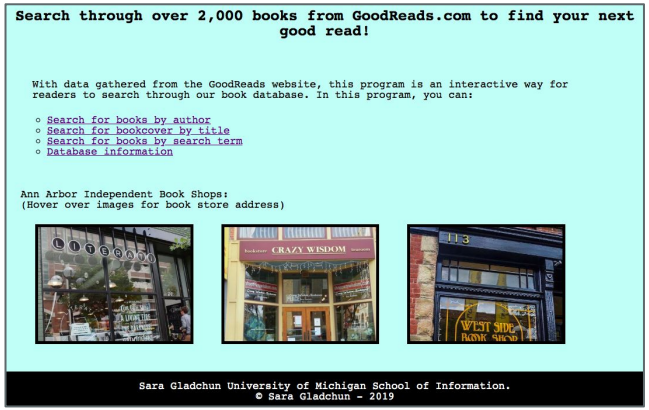
author-form.html

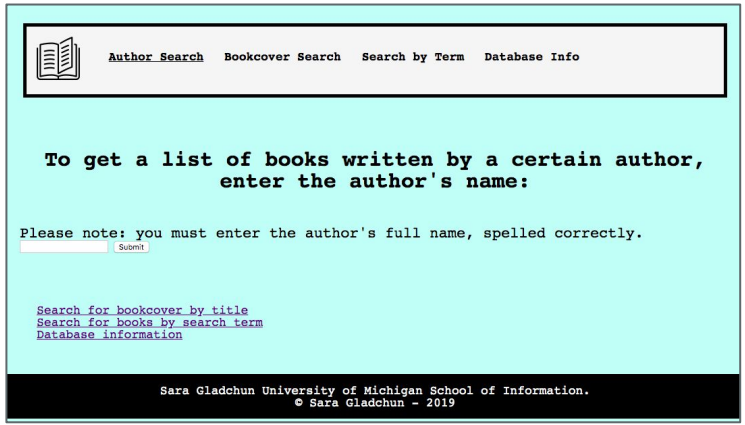
```
<h1> To get a list of books written by a certain author, enter the author's name: </h1>
<p>Please note: you must enter the author's full name, spelled correctly.</p>

<form method="POST">
    <input name="text">
    <input type="submit">
</form>
<a href="{{ url_for('cover_form')}}">Search for bookcover by title</a><br>
<a href="{{ url_for('search_form')}}">Search for books by search term</a><br>
<a href="{{ url_for('index')}}">Database information</a>
```

After



- Added CSS and styling
- Adding divs, classes, and IDs
- Unified background color
- Unified header, footer, or nav bar





Updates Made

HTML

- Added divs and classes to better style the page
- Added formatting to the page
- Added a unified header, navigation, and footer to each page
- Added bookstore locations and images to the homepage to create a more interesting user experience

Backend

- Added ISBN to database for future API use

CSS

- Background color (accessibility checked on: https://webaim.org/resources/contrastchecker/)
- Changed font for the website
- Unifying header, footer, and navigation
- Adding formatting to the homepage
- CSS flip cards for homepage images
- Clickable icon to get back to homepage (previously no way to return to the homepage - better UX)
- Added indicator for current page and icon to return to the homepage

Process Overview

1. Sketched out general idea for an updated layout for the web page
2. Created a mapping for the user experience through the page
3. Duplicated backend of the project to make changes to the templates and static files for HTML and CSS
4. Made updates to the HTML and added CSS
5. Tested page for accessibility
6. Made updates for accessibility
7. Tested page for functionality
8. Tested web page with 3 users for user experience feedback

Biggest Challenges:

1. Time constraints (12-15 hours only allowed enough time for certain changes)
2. Getting the layout for the navigation to layout properly with the icon (due to time constraints I was only able to do this for the website media query - not mobile or responsive)
3. Working with Javascript API - added the ISBN to the database to later work with the Google Books API (unable to complete)
4. Accessibility for my forms - the forms I had used for my python project were insufficient for accessibility so I needed to make them more accessible

Next Steps

Backend:

- Add ISBN to class objects and database
- Write API call to Google Books API or amazon API to query ordering and pricing information to website
- Write API in order to search nearby bookstores (rather than default Ann Arbor bookstores) and get image and address (Google Places API?)

Frontend:

- Make responsive web pages
  - Mobile
  - Larger screen
  - Reduced Motion
- JavaScript API to pull image from GoodReads Instagram and/or from Google Books API
- Add a grid layout to the page in order to have related links and images on the side of the pages