

Design Elements Report

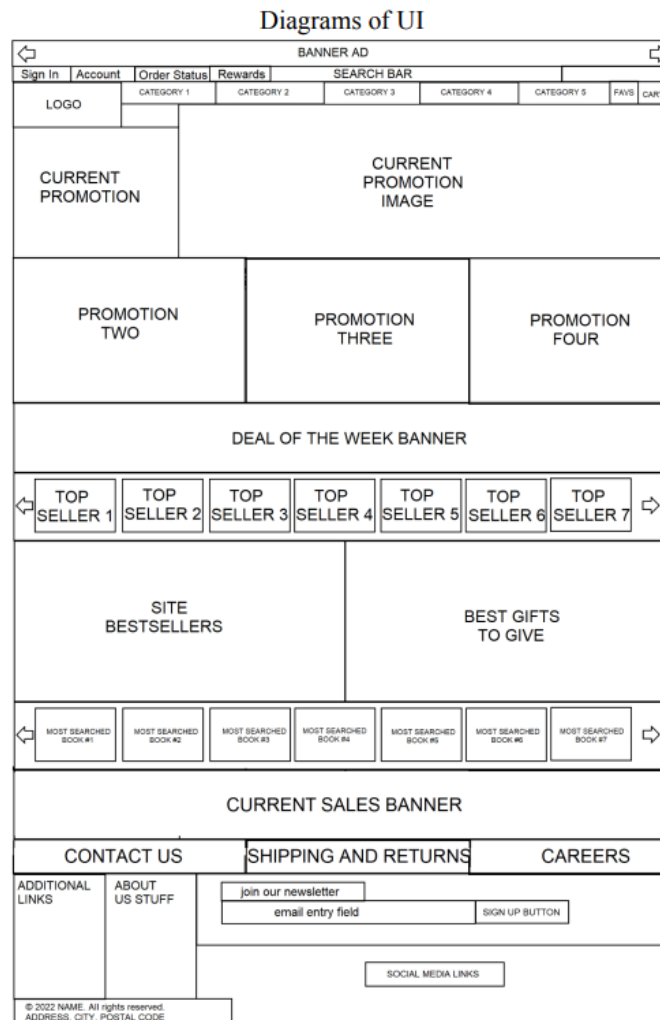
Design Criteria

This project's design must be modularized as much as possible to make developing the website easier. The website has to contain a shopping cart feature, as well as a checkout page and a user login. If an administrator logs in, they should be able to edit the inventory of books.

Description of the Design

The design of the website will be as follows:

There will be a main homepage, where there are featured books and links to each category. There will also be a search bar on the top right of the page. If you enter a query into the search bar, the user will be redirected to a results page, where the search results will appear. Once a customer would like to add something to their cart, they can select the 'add to cart' button. Upon checkout, the user can enter their billing information. After successful checkout, the user will be redirected to an email confirmation page. Below is a sketch of the homepage interface:



All the information related to the selection of books will be in a SQL database. In this database, the name, author, and other relevant information is included for each book. Most importantly, each book is characterized by a unique id number. When viewing a book, this id is used to let the php code know which book's information to display. The scraper to obtain the books and their accompanying information is written in python, which is completely separate from the php code used to read from the database.

Classes

There are 6 main classes in our design: the database, books, shopping cart, inventory, shipping and handling, and account. Having multiple classes in our design allows for modularity. Making elements less intertwined results in a more elegant product. Only when needed will the classes interact with each other to run functions or access variables. We used the separation of concerns principle to guide our project.

Recommendations and Trade-Offs

We chose to approach our inventory using SQL and Php. The advantage of doing this was that our team members have already been introduced to these languages and tools, so implementing our ideas did not have that large of a learning curve. We chose to web scrape the book information so that we will not be limited to however many books we could hardcode ourselves. An advantage to using a web scraper was that we could have real books and pdfs on our website.

The tradeoff of using a web scraper was that our team did not have enough time to fully implement it. We have the working code, just have not used the finished database for the Php retrieval. Another trade-off that we made was using copyrighted books for this project. The legality of having a website host copyrighted books' PDFs are not clear. Although it is not illegal as long as we are not distributing it. It may be deemed unethical to deploy the completed version of this project to the web.